

Article

# SCOPE-Based Emulators for Fast Generation of Synthetic Canopy Reflectance and Sun-Induced Fluorescence Spectra

Jochem Verrelst<sup>1,\*</sup>, Juan Pablo Rivera Caicedo<sup>1,2</sup>, Jordi Muñoz-Mari<sup>1</sup>, Gustau Camps-Valls<sup>1</sup> and José Moreno<sup>1</sup>

<sup>1</sup> Image Processing Laboratory (IPL), Parc Científic, Universitat de València, 46980 Paterna, Spain; jprivera@conacyt.mx (J.P.R.C.); jordi.munoz@uv.es (J.M.-M.); Gustau.Camps@uv.es (G.C.-V.); jose.moreno@uv.es (J.M.)

<sup>2</sup> CONACyT-UAN, Secretaría de Investigación y Posgrado, Universidad Autónoma de Nayarit, Ciudad de la Cultura Amado Nervo, Tepic CP. 63155, Nayarit, Mexico

\* Correspondence: jochem.verrelst@uv.es; Tel.: +34-96-354-40-67; Fax: +34-96-354-32-61

Received: 24 July 2017; Accepted: 1 September 2017; Published: 6 September 2017

**Abstract:** Progress in advanced radiative transfer models (RTMs) led to an improved understanding of reflectance ( $R$ ) and sun-induced chlorophyll fluorescence ( $SIF$ ) emission throughout the leaf and canopy. Among advanced canopy RTMs that have been recently modified to deliver  $SIF$  spectral outputs are the energy balance model SCOPE and the 3D models DART and FLIGHT. The downside of these RTMs is that they are computationally expensive, which makes them impractical in routine processing, such as scene generation and retrieval applications. To bypass their computational burden, a computationally effective technique has been proposed by only using a limited number of model runs, called emulation. The idea of emulation is approximating the original RTM by a surrogate machine learning model with low computation time. However, a concern is whether the emulator reaches sufficient accuracy. To this end, we analyzed key aspects of emulator development that may impact the precision of emulating SCOPE-like  $R$  and  $SIF$  spectra, being: (1) type of machine learning, (2) type of dimensionality reduction (DR) method, and (3) number of components and lookup table (LUT) size. The machine learning family of Gaussian processes regression and neural networks were found best suited to function as emulators. The classical principal component analysis (PCA) remains a robust DR method, but the number of components needs to be optimized depending on the complexity of the spectral data. Based on a small Latin hypercube sampling LUT of 500 samples (70% used for training) covering a selection of SCOPE input variables, the best-performing emulators can reconstruct any combination for the selected SCOPE input variables with relative errors along the spectral range below 2% for  $R$  and 4% for  $SIF$ . That is sufficient for a precise reconstruction for the large majority of possible combinations, and errors can be further reduced when increasing LUT size for training. As a proof of concept, we imported the best-performing emulators into a newly developed Automated Scene Generator Module (A-SGM) to generate a  $R$  and  $SIF$  synthetic scene of a vegetated surface. Using emulators as alternative of SCOPE reduced the processing time from the order of days to the order of minutes while preserving sufficient accuracy.

**Keywords:** spectroscopy; sun-induced fluorescence; radiative transfer modelling; emulation; SCOPE; machine learning; scene generation

## 1. Introduction

While the exploitation of imaging spectroscopy data for quantifying vegetation properties has long been restricted to top-of-canopy (TOC) reflectance ( $R$ ) (see [1–3] for reviews), there is a growing

interest in exploiting canopy-leaving sun-induced-chlorophyll-fluorescence (*SIF*) emissions to enable quantifying the physiological status of vegetation more directly [4–6]. Specifically, the forthcoming European Space Agency’s 8th Earth Explorer FLEX (FLuorescence EXplorer) is dedicated to exploit *SIF* emissions from terrestrial vegetation [7] on a global scale and link it to both plant productivity and plant stress. The rationale of the FLEX mission is based on sound scientific foundations in plant physiology [8–10] and radiative transfer principles [11–14] that describe the propagation of the *SIF* signal through the cell, leaf, canopy and atmosphere. Progress in both domains has led to a recent boost in the development of physically-based radiative transfer models (RTMs) of vegetation canopies with mechanisms of *SIF* propagation throughout the leaf and canopy [15,16]. For instance, significant progress has been achieved with the advanced soil-vegetation-atmosphere-transfer (SVAT) model Soil-Canopy Observation of Photochemistry and Energy fluxes (SCOPE) [17]. SCOPE is a coupled *SIF*-photosynthesis RTM that simulates photosynthesis, radiative transfer in the leaf and canopy, and surface energy balance. The basic principle is that *SIF*-photosynthesis mechanisms are first calculated at the leaf scale and then propagated throughout the canopy. Based on this principle, similar *SIF* leaf-to-canopy mechanisms have been recently introduced into even more advanced RTMs that explicitly account for canopy structure, such as DART [18] and FLIGHT [19].

These advanced RTMs can potentially function as virtual laboratories to study the role of biochemical, leaf and canopy variables on the spectral outputs such as *R* and *SIF* [20,21]. A major bottleneck using advanced RTMs, however, is that they are computationally expensive, which makes them impractical in routine processing such as retrieval applications. In order to bypass their computational burden, a computationally effective technique has been proposed by only using a limited number of computer code runs. The core idea is approximating the original deterministic model by a surrogate statistical learning model, also referred to as a meta-model or *emulator*. When an accurate emulator has been developed, it can then approximate the functioning of the original model—and this at a tiny fraction of the original computational burden—and be readily applied in tedious processing applications. Although longer known in computer sciences [22,23], emulation (or surrogate modelling) of a deterministic model (e.g., a RTM) is very new in optical remote sensing science and especially in the field of imaging spectroscopy [24]. Once having a fast RTM emulator developed, it opens a variety of new opportunities in replacing computationally expensive RTMs in spectroscopy and remote sensing applications. For instance, the emulator can be applied to all kinds of repetitive tasks that were never possible before due to the heavy computational load of the advanced RTM, such as global sensitivity analysis [25], scene generation and retrieval (inversion) schemes.

Progressing along this line, the emulation technique has recently been made user-friendly into a so-called Emulator toolbox [26], which forms part of the ARTMO (automated radiative transfer models operator) software package [27]. ARTMO streamlines the running of a suite of RTMs at the leaf and canopy scale, including SCOPE. The Emulator toolbox is equipped with a diversity of machine learning regression algorithms (MLRAs) and enables to develop and optimize statistical models and apply them as emulators. These software developments greatly facilitate the training and validation of emulators that can mimic advanced RTMs. Accurate emulators can then replace the original RTM in further processing.

When it comes to importing an emulator into tedious processing tasks, a critical aspect to be clarified is to be confident about its precision in approximating the original RTM. After all, the emulator is essentially a statistical simplification of the original model. Given that an RTM is already a simplification of reality, there may be a risk of running into oversimplification, or even producing unrealistic outputs. Consequently, by introducing emulators into a processing scheme, one should not only strive for the development of a fast emulator, but probably more importantly, for an emulator that maximizes accuracy, efficiency and trustworthiness. Nevertheless, the development of an emulator that is both fast and accurate is not a trivial task. In fact, various factors determine the performance of the emulator, with most importantly the complexity of the to be emulated RTM and the used machine learning algorithm [25,26]. At the same time, spending all efforts in maximizing accuracy may go at

the expense of processing speed. It thus implies that, depending on the application, a trade-off has to be made between the degree of accuracy and gain in processing speed.

Henceforth, a systematic evaluation of the key aspects that play a role in the performance of the emulator is urgently demanded. Not in the least when data processing relies on advanced RTMs such as *SIF* modelling. Aspects to be assessed include: (1) used machine learning method, (2) applied dimensionality reduction strategy, and (3) lookup table (LUT) size to train the machine learning methods. They have to be analyzed and consolidated before reaching conclusions whether emulators offer truly an attractive alternative over the original RTM. Specifically, SCOPE is currently used as baseline model in the FLEX End-to-End simulator [28] for delivering of *R* and *SIF* spectral outputs, and in the development of *SIF* retrieval methods [29] and subsequent interpretation towards photosynthetic activity [21,30–32]. Thus, if evaluated as sufficiently accurate, an emulated version of SCOPE will greatly speed up all these processes. This does not mean that the complete SCOPE model has to be emulated; for instance not all variables play a role in shaping *R* and *SIF* spectral outputs [20], and some variables can be kept fixed because typically known, e.g., sun-target-sensory geometry. Hence, an emulator can be customized as a simplified version of the original RTM for a specific application.

In light of the above, the main objective of this study is to assess the degree of accuracy that can be preserved in emulating SCOPE *R* and *SIF* outputs given a subset of input variables, and what gain in processing speed can be achieved. This study breaks down into the following two sub-objectives. The first sub-objective refers to the selection of the right statistical learning method by assessing the role of: (1) machine learning algorithm and (2) dimensionality reduction strategy. Once having the best-performing machine learning method identified, the second sub-objective strives for further model optimization and refers to assessing the role of: (3) number of components, and additionally (4) the training LUT size. As a practical application, the best-performing emulator will subsequently be applied to the rendering of a *R* and *SIF* emission scene as would be acquired by an imaging spectrometer over a vegetated surface (after atmospheric corrections). A discussion on identified opportunities for fast spectroscopic data processing applications will close this paper.

## 2. Emulation Theory

Emulation is a technique used to estimate model simulations when the computer code under investigation is too computationally costly to be run enough times [23]. The concept of developing emulators based on statistical learning have already been applied for the last few decades in the climate and environmental modeling communities e.g., [33–40]. These emulators have in common that they were developed from adaptive, flexible machine learning regression algorithms (MLRAs) [41]. Earlier examples of MLRAs that can successfully function as emulator include Gaussian processes regression (GPR) and neural networks (NN) [24,25,42]. The basic idea is that the emulator uses a limited number of simulator runs, i.e., input-output pairs (corresponding to training samples), to infer the values of the complex simulator output given a yet-unseen input configuration. These training data pairs should ideally cover the multidimensional input space using a space-filling sampling algorithm, e.g., Latin hypercube sampling [43]. Once the emulator is built, it is not necessary to perform any additional runs with the model; the emulator computes the output that is otherwise generated by the simulator [23]. Hence, emulators are statistical models that can generalize the input-output relations from a subset of examples generated by RTMs to unseen data. Note that building an emulator is essentially nothing more than building an advanced regression model as typically done for retrieval applications (see pioneering works using NNs [44,45] and also more recent MLRAs e.g., [46–48]), but then in reversed order: whereas a retrieval model converts spectral data (e.g., *R*) input into one or more output biophysical variables, an emulator converts input biophysical variables into output spectral data.

When it comes to emulating RTM spectral output, however, the challenge lies in delivering a full spectrum, i.e., predicting multiple spectral bands. This is an additional difficulty compared to

standard emulators that only deliver one or few outputs [49]. It bears the consequence that the MLRA should be able to generate multiple outputs to be able reconstructing a full spectral profile. This is not a trivial task. The full, contiguous spectral profile typically consists of over 2000 bands when binned to 1 nm resolution. Only very few regression models possess the capability to generate multi-outputs; the most common example is NN. Moreover, to train a complex multi-output statistical model with the capability to generate so many output bands can take considerable training time, especially when using NN [26]. A workaround solution has to be developed that enables the MLRA to cope with large, spectroscopic datasets (e.g., *R*, *SIF*). An efficient solution is to take advantage of the so-called ‘curse of spectral redundancy’, i.e., the Hughes phenomenon [50]. Spectroscopic data typically possess a great deal of collinearity, which implies that data can be converted to a lower-dimensional space through dimensionality reduction (DR) techniques. Accordingly, spectroscopic data can be converted into components, which are only a fraction of the original amount of bands. As such, the multi-output problem is greatly reduced to a number of components that preserve the spectral information content. Afterwards the components can then again be reconstructed into spectral data. The classical principal component analysis (PCA) [51] is the most popular DR method, but also alternative DR methods can be considered given they enable reconstruction of the original spectral data. Thus, by first applying a DR method the contiguous spectral data is reduced into a given number of features (components). By subsequently looping over the single components, multiple models can be trained by a single-output MLRA. Afterwards the signal is again reconstructed. Although the iterative training takes some computational time, it goes considerably faster than training band-by-band a contiguous spectral dataset, and makes the problem better conditioned [24–26,39]. More generally, by introducing a DR step it allows converting single-output MLRAs into multi-output algorithms, i.e., meaning that in principle any multivariate regression algorithm can function as emulator, although probably only those within the family of machine learning are sufficiently adaptive to enable accurately approximating the behavior of a RTM.

### 2.1. Machine Learning Regression Algorithms (MLRAs)

To enable developing an RTM-based emulator, the first step involves building a statistically-based representation (i.e., an emulator) of the RTM from a set of training data points derived from runs of the actual model under study. These training data pairs should ideally cover the multidimensional input space using a space-filling algorithm. The second stage uses the emulator built in the first step to compute the output that is otherwise generated by the RTM [23]. Based on the literature review above and an earlier comparison study [25,26], the following six MLRAs potentially serve as powerful regression methods to function as accurate emulators, being: (1) random forest (RF); (2) artificial neural networks (NN); (3) support vector regression (SVR); (4) kernel ridge regression (KRR), also known as least squares support vector machine; (5) Gaussian processes regression (GPR); and (6) its variational heteroscedastic variant (VHGPR). These MLRAs can be categorized into three groups: (1) decision trees, (2) neural networks, and (3) kernel methods, and are briefly outlined below.

Decision tree learning is based on decision tree predictive modeling. A decision tree is based on a set of hierarchical connected nodes. Each node represents a linear decision based on a specific input feature. A classical decision tree algorithm cannot cope with strong non-linear input-output transfer functions. In that case, a combination of decision trees can improve results, such as random forests [52].

Artificial neural networks (NNs) are essentially fully connected layered structures of artificial neurons (AN) [53]. A NN is a (potentially fully) connected structure of neurons organized in layers. Neurons of different layers are interconnected with the corresponding links (weights). Training a NN implies selecting a structure (number of hidden layers and nodes per layer), initialize the weights, shape of the nonlinearity, learning rate, and regularization parameters to prevent overfitting. The selection of a training algorithm and the loss function both have an impact on the final model. In this work, we used the standard multi-layer perceptron, which is a fully-connected network. We selected just one

hidden layer of neurons. We optimized the NN structure using the Levenberg-Marquardt learning algorithm with a squared loss function.

Kernel methods in machine learning owe their name to the use of kernel functions. Kernels quantify similarities between input samples of a dataset [54]. Similarity reproduces a linear dot product (scalar) computed in a possibly higher dimensional feature space, yet without ever computing the data location in the feature space. The following two methods are gaining increasing attention: (1) support vector regression (SVR) [55], (2) Kernel ridge regression (KRR), also known as least squares support vector machines [56], and (3) Gaussian processes regression (GPR), based on Gaussian processes, which generalize Gaussian probability distributions in function spaces [57]. The Standard Gaussian processes model observations' noise is treated as constant throughout input space. This is often a too restrictive assumption, but one that is needed for GP inference to be tractable. As an alternative, the variational approximation allows accurate inference in heteroscedastic GPs (i.e., under input-dependent noise conditions), i.e., (4) variational heteroscedastic GPR (VHGPR) [58]. For all kernel methods we used a standard radial basis function kernel.

Each of these advanced regression methods are popular in various application domains thanks to their relatively fast training, good performance and robustness to the overfitting problem. A more detailed comprehensive description of the methods is given in [48,59] and a MATLAB implementation in [60].

## 2.2. Dimensionality Reduction Algorithms

An important aspect in the reconstruction of RTM spectral output is that the MLRA is used in combination with a DR method. The MLRA first predicts the DR-reduced spectra, and the inverse of the DR transformation is subsequently applied to obtain the reconstructed output spectra. Extracting meaningful components from multidimensional data is typically done using PCA. The PCA step consists on projecting the RTM output spectra onto the first PCA components,  $p \ll B$ , where  $p$  is the number of selected components (usually one typically takes enough components to ensure that at least 99.99% of variance is retained), and  $B$  is the number of original bands of the spectra. Solving the PCA is done by obtaining the eigenvectors and eigenvalues of the estimated covariance matrix of the spectral inputs  $X$ . The eigenvectors matrix,  $U$ , is then used as a projection matrix that allows to obtain the so-called X-scores, simply by  $S = U \cdot X$ . As  $U$  is an orthogonal matrix the reconstruction of  $X$  given the scores is obtained by  $X = U^T \cdot S$ . Hence, the spectra is reconstructed.

Also more advanced DR methods have been considered (e.g., kernelized DR methods), but they did not allow reconstruction of the spectral output. Yet a DR method that can be applied in the same way as PCA involves partial least squares (PLS) [61]. While PCA disregards the target data (i.e., RTM input variables) and exploits correlations between the features (i.e., RTM output spectra) to maximize the variance of the projections, PLS looks for projections that maximize, respectively, the covariance and the correlation between the features and the target data. Having now the possibility to couple multiple MLRAs with either PCA or PLS, the next step would be assessing the performance of ensembles of emulation strategies in order to consolidate an optimized emulator.

## 3. ARTMO Modules: Emulator Toolbox, SCOPE and A-SGM

The complete assessment was undertaken within the in-house developed ARTMO software package [27]. In short, ARTMO consists of a suite of leaf and canopy RTMs including SCOPE, retrieval toolboxes, and recently an Emulator toolbox was added [26]. In the Emulator toolbox, several of those MLRAs can be trained by RTM-generated LUTs, whereby biophysical variables are used as input in the regression model, and spectral data is generated as output. For this study, the Emulator toolbox has been expanded with new machine learning algorithms (i.e., SVR, RF, VHGPR) an alternative DR method (i.e., PLS) and goodness-of-fit indicators per wavelength (v. 1.07). Additionally, as a new ARTMO toolbox, the automated Scene Generated Module (A-SGM) is here for the first time presented

(v. 1.11). ARTMO is freely downloadable at <http://ipl.uv.es/artmo/>. SCOPE and A-SGM are briefly described below.

### 3.1. SCOPE

The coupled fluorescence-photosynthesis model SCOPE was used to develop and evaluate emulators. SCOPE simulates photosynthesis, radiative transfer in the leaf and canopy, and surface energy balance [17,62]. The model recently became a virtual laboratory for studies on surface energy balance [63], remote sensing thermal infrared measurements [64], and *SIF* photosynthesis studies [20,21,30,65]. SCOPE combines a number of radiative transfer models (RTM): one for the leaf, Fluspect [16] and three for the canopy: RTM<sub>o</sub>, RTM<sub>t</sub> and RTM<sub>f</sub> [62]. Fluspect is an extension of the widely used PROSPECT model [66], which simulates the radiative transfer of incident light and emitted *SIF* within the leaf. The outputs are the reflectance and transmittance spectra, and the excitation-emission matrices for both sides of the leaf. Routines for photochemical quenching (PQ) and NPQ of the *SIF* are used together with the output of the Fluspect model to simulate the *SIF* emission per leaf angle and leaf layer. The emitted *SIF* then is used in the module RTM<sub>f</sub>, which simulates scattering and absorption of *SIF* within the canopy. The biochemical model for PQ and NPQ requires leaf temperatures as input, calculated from the energy balance module and the RTM<sub>t</sub> module for radiative transfer of thermal radiation. SCOPE includes a number of biochemical routines for PQ and NPQ. We used the model of balance [17], which is a combination of a photosynthesis exchange models for C3 and C4 vegetation [9,10], the gas exchange model of Ball-Berry model [8], and an empirical parameterization of the relationship between PQ and NPQ. We selected the parameterization calibrated from data as provided in [17]. The distribution of solar irradiance over leaves in the canopy is simulated with the model RTM<sub>o</sub>: a four-stream radiative transfer model. The four simulated fluxes include direct, diffuse upward, diffuse downward and flux in the observation direction. In order to simulate *SIF* outputs, SCOPE requires a series of meteorological parameters and four classes of variables: (1) vegetation structure variables: canopy height, leaf orientation distribution and leaf area index (*LAI*); (2) leaf biophysical variables: chlorophyll content (*Cab*), dry matter content (*Cm*), leaf equivalent water thickness (*Cw*), senescent material (*Cs*); (3) optical variables: *R* of soil in the visible, near-infrared and thermal bands, and vegetation (thermal) emissivity; and (4) plant physiological variables. SCOPE v. 1.61 is used here.

### 3.2. Automated Scene Generator Module (A-SGM)

As a practical application, here we use SCOPE-based emulators to generate synthetic hyperspectral scenes. In preparation of the FLEX mission a so-called End-to-End simulator (FLEX-E) has recently been developed [28]. FLEX-E enables to reproduce all aspects of a satellite mission, including platform orbit/attitude, synthetic scene radiative transfer generation, sensor behavior, ground image processing, and product evaluation. Specifically, the Scene Generator Module (SGM) [67] propagates solar radiation through the canopy and atmosphere, simulating on a pixel-by-pixel basis the target scenes for FLEX and Sentinel-3 sensors. The scenes are defined according to key biophysical, atmospheric, and topographic input parameters and produces TOC *R* and *SIF* and top-of-atmosphere (TOA) radiance spectra. Given that the generation of synthetic scenes by RTMs is an extremely time-consuming process, we explored the possibility to replace the computationally-expensive RTMs (i.e., SCOPE, MODTRAN) by customized emulators. In this way, it would allow to rapidly generate large or multiple scenes, including time series of images, e.g., as would be observed by imaging spectrometers such as FLEX. To do so, within the ARTMO framework an automated SGM (A-SGM) toolbox has been developed. A-SGM is automated in a sense that the user only has to go through a few graphical user interfaces (GUIs) to define the scene settings, land cover class settings, and optionally definition of temporal profiles. Once these steps are defined, input images (e.g., spatially distributed biophysical variables) and output scenes (e.g., *R* or *SIF*) are automatically generated. The A-SGM is a simplified version of the FLEX-E SGM in a sense that it currently only

generates TOC spectra (and no TOA radiance data). In short, A-SGM enables to: (1) generate simulated scenes according to the band settings of a sensor based on coupled leaf and canopy models such as SCOPE, or based on emulators; (2) define multiple land cover classes (e.g., plant functional types) and configure them with RTM/emulator input variables. Land cover maps can be either imported or arbitrarily generated; (3) define land cover classes with various spatial patterns of input variables (4) generate time series of the simulated scenes based on temporal trends of the input variables; and (5) store and visualize input and output maps, and provides simple statistics of generated products, both in spectral, spatial and temporal domains. More information can be found in [28,67].

#### 4. Experimental Setup

Here we outline the experimental setup for running the emulation experiments and assessment. To start with, the purpose was not to develop emulators that cover all SCOPE input variables since various variables play no role in shaping either *R* or *SIF* spectral outputs, or are considered as known (e.g., geometry). Based on an earlier global sensitivity analysis [20] only driving vegetation variables and soil moisture content were selected (Table 1). A LUT was first generated by means of Latin hypercube sampling (LHS) within the RTM variable space with minimum and maximum boundaries as given in Table 1. A LHS of training data is preferred, as LHS covers the full parameter space, and thus, in principle, assures that the developed emulator will be able to reconstruct correct spectral output for any possible combination of input variables. A sampling size was sought for that balances between striving for high accuracies while keeping computational burden to the minimum. Earlier emulation studies suggested that using just 400 samples would suffice to function as emulator [33,68]. On the other hand, a larger LUT size may lead to the development of a more accurate emulator [26]. Nevertheless, increasing input simulations goes at a computational cost. As an acceptable compromise, a LUT size of 500 samples was built as default situation.

**Table 1.** Range of vegetation input variables used to establish the SCOPE 500 LUT according to Latin Hypercube sampling. The other SCOPE variables are kept fixed to the SCOPE default values as provided in Appendix A (Table A1). \* A LAI of 10 may be too extreme, a LAI until 6 would be more common.

Model Variables		Units	Minimum	Maximum
<b>Leaf variables:</b>				
N	Leaf structure index	unitless	1.0	4.0
Cw	Leaf water content	[cm]	0.001	0.05
Cab	Leaf chlorophyll content	[ $\mu\text{g}/\text{cm}^2$ ]	0.1	100
Cm	Leaf dry matter content	[ $\text{g}/\text{cm}^2$ ]	0.0001	0.05
Cs	Senescent material	unitless	0.0001	0.3
<b>Canopy variables:</b>				
LAI	Leaf area index	[ $\text{m}^2/\text{m}^2$ ]	0.01	10 *
hc	plant height	[m]	0	2
SMC	volumetric soil moisture content	unitless	0.01	0.7

The processing time of the SCOPE 500 LUT was 37 min on a contemporary computer (Windows-64 OS, i7-4550 CPU 1.50 GHz, 8GB RAM). Because SCOPE delivers all types of outputs, here only the directional *R* and the *SIF* spectral outputs are used for developing emulators, i.e., two LUTs (same inputs but different outputs) are used in subsequent analysis. Each LUT was subsequently passed to the Emulator toolbox to evaluate the different MLRAs. The classical PCA was first applied. Although for each of the MLRAs the first 5 components explained already 99.96% of the variance, depending on the complexity of the output spectra, developing the regression models with less than 10 components led to inaccurate *R* reconstructions. Since *SIF* output is a smooth 650–850 nm two-peak broadband signal, 5 components was considered sufficient to start the analysis. However, for emulating *R* spectra more precise profiles were achieved with having more components added

in the regression algorithms. However, looping over more components slows down processing time. Therefore, 20 components (i.e., 100% explained variance) were evaluated as an acceptable trade-off between accuracy and processing time to the start with the *R* emulation analysis. With this default setup, the following aspects are systematically analyzed for emulating *R* and *SIF* spectra:

1. The selected MLRAs are first analyzed for the default training LUT of 500 samples and using 20 PCA for emulating *R* spectra and 5 PCA for emulating *SIF* spectra.
2. For the best-performing MLRA, the roles of PCA and PLS are analyzed.
3. For the best-performing MLRA and DR method, the following number of components are evaluated: 10, 20, 30 and 40 components for *R* spectra, and 2 up to 10 components for *SIF* spectra.
4. Additionally, the role of LUT size is analyzed for the best-performing emulator. A new LUT of 1000 samples according to LHS has been generated and evaluated.

Finally the best evaluated emulators are used in the rendering of a synthetic hyperspectral scene of TOC *R* and canopy-leaving *SIF* as would be observed by an imaging spectrometer overflying a vegetated surface (without atmospheric effects).

#### 4.1. Emulation Validation

Since emulators only produce an approximation of the original model, it is important to realize that such an approximation introduces a source of uncertainty referred to as “code uncertainty” associated with the emulator [23]. Therefore, validation of the generated model is an important step in order to quantify the emulator’s degree of accuracy. To validate the accuracy of the emulator, part of the original data is kept aside as reference dataset. Various training/validation sampling design strategies are possible with the Emulator toolbox. In an attempt to keep processing fast, same as in [25] only a single split was applied using 70% samples for training and the remaining 30% for validation. The Emulator toolbox further provides various tools to validate the developed emulators. It analyzes the validation accuracy of each emulator by calculating the root-mean-square-error (RMSE) and the normalized RMSE (NRMSE) [%] difference between emulated spectra and validation RTM spectra per wavelength ( $\lambda$ ) and then averaged over the full spectral range:

$$RMSE_{\lambda} = \frac{\sum_{\lambda=1}^b \sqrt{\frac{\sum_{n=1}^s (e_{\lambda,n} - r_{\lambda,n})^2}{s}}}{b} \quad (1)$$

$$NRMSE_{\lambda} [\%] = \frac{\sum_{\lambda=1}^b \left( \frac{\sqrt{\frac{\sum_{n=1}^s (e_{\lambda,n} - r_{\lambda,n})^2}{s}}}{\max(\sum_{\lambda=1}^b r_{\lambda,n}) - \min(\sum_{\lambda=1}^b r_{\lambda,n})} \right)}{b} 100 \quad (2)$$

where  $b$  is the number of bands,  $s$  the number of samples,  $e$  the emulated spectral value and  $r$  the RTM spectral value. The processing time of model training and validation is also tracked. In order to inspect the emulation accuracy along the spectral range, the toolbox further offers some visualization tools such as plotting the RTM vs. emulated output spectra and plotting the residuals, i.e., the difference between RTM-generated and emulated-generated spectra in absolute or relative terms. These plottings can also be summarized into general statistics, e.g., mean, standard deviation (SD), min-max boundaries and percentiles. Another option involves plotting the goodness-of-fit statistics as a function of wavelength, i.e., RMSE, NRMSE, coefficient of variation:  $R^2$ . By plotting the errors over the full spectral range (e.g., RMSE or NRMSE), the achievable accuracy in reproducing spectral output can be quickly compared.

#### 4.2. Scene Definition

As a proof of concept to exemplify the utility of emulators, the best-performing *R* and *SIF* emulators were imported into the A-SGM for a scene generation. Based on vegetation properties



of three arbitrarily defined vegetation classes, the emulators generated artificial scenes of  $334 \times 334$  (111,556 pixels) of *R* and *SIF* outputs at 1 nm spectral resolution. For these classes, the following variables have been ranging by giving a mean and a SD around it (Table 2). The spatial patterns of the land cover classes are also arbitrarily generated, and within each vegetation class the input variables are uniformly randomly distributed.

**Table 2.** Range of vegetation input variables used to fill the three synthetic land cover classes.

Model Variables	Units	Vegetation Class 1 Mean (% SD)	Vegetation Class 2 Mean (% SD)	Vegetation Class 3 Mean (% SD)
<b>Leaf variables:</b>				
N	unitless	1.2 (30%)	1.4 (30%)	1.1 (10%)
Cw	[cm]	0.02 (30%)	0.01 (30%)	0.01 (10%)
Cab	[ $\mu\text{g}/\text{cm}^2$ ]	40 (30%)	50 (30%)	30 (10%)
Cm	[ $\text{g}/\text{cm}^2$ ]	0.02 (30%)	0.03 (30%)	0.03 (10%)
Cs	unitless	0.001 (30%)	0.001 (30%)	0.001 (10%)
<b>Canopy variables:</b>				
LAI	[ $\text{m}^2/\text{m}^2$ ]	3 (30%)	5 (30%)	2.5 (10%)
hc	[m]	0.3 (30%)	2 (30%)	0.3 (10%)
SMC	unitless	0.02 (30%)	0.02 (30%)	0.02 (30%)

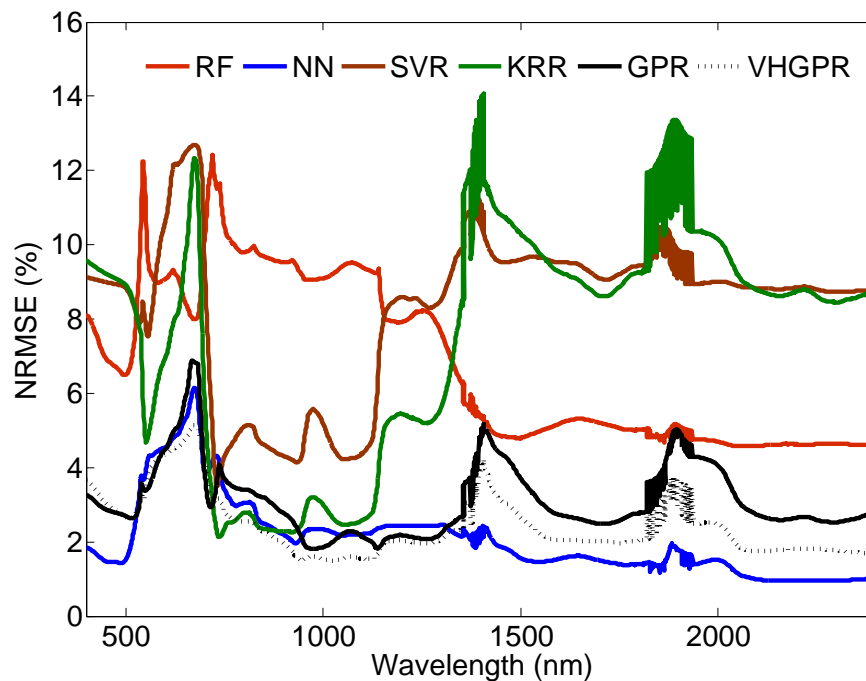
## 5. Results

Results are organized as follows. First the key aspects that play a role in the emulation of *R* spectra are systematically analyzed. Then the analysis is repeated for the emulation of *SIF* spectra. Finally the best-performing emulators are used in the rendering of a synthetic hyperspectral scene of TOC *R* and canopy-leaving *SIF*.

### 5.1. SCOPE Reflectance (*R*) Spectra

#### 5.1.1. On the Role of Used MLRA

The role of the selected MLRAs on their ability of emulating *R* spectra are first analyzed for the default situation of trained by a LUT of 500 samples according to a 70/30% training/validation distribution and a 20 PCA conversion. Validation results are compared by plotting the relative errors (NRMSE) across the spectral range (Figure 1) and overview statistics and processing time are given in Table 3. The figure suggests the following: NN and the two GPR methods emulate substantially more accurate *R* spectra than the other MLRAs. Overall, relative errors stay well below 5%, although in the red to red-edge region (620–750 nm) accuracies are poorer. The GPR methods face additional difficulties in the water absorption regions (around 1400 and 1900 nm). Although GPR performs somewhat poorer than NN, it still deserves special attention. Table 3 shows that GPR accuracies are only slightly worse than NN. However, processing time halved, which refers mostly to training time, since for all methods applying the emulator for validation took place almost instantly. KRR only delivers acceptable accuracies within the NIR 750–1100 nm region, yet KRR is remarkable because mathematically simpler than GPR, which results into tremendously faster processing time, i.e., on the order of seconds. RF and SVR perform considerably poorer and can therefore be discarded to function as emulator.



**Figure 1.** SCOPE NRMSE [%] results for the MLRAs validation for 30% of 500 SCOPE  $R$  outputs using 20 PCA conversion.

**Table 3.** MLRA emulators validation results ( $RMSE_{\lambda}$ ,  $NRMSE_{\lambda}$ ) and training and validation processing time (s: seconds) for 30% of 500 SCOPE  $R$  outputs and using 20 PCA conversion).

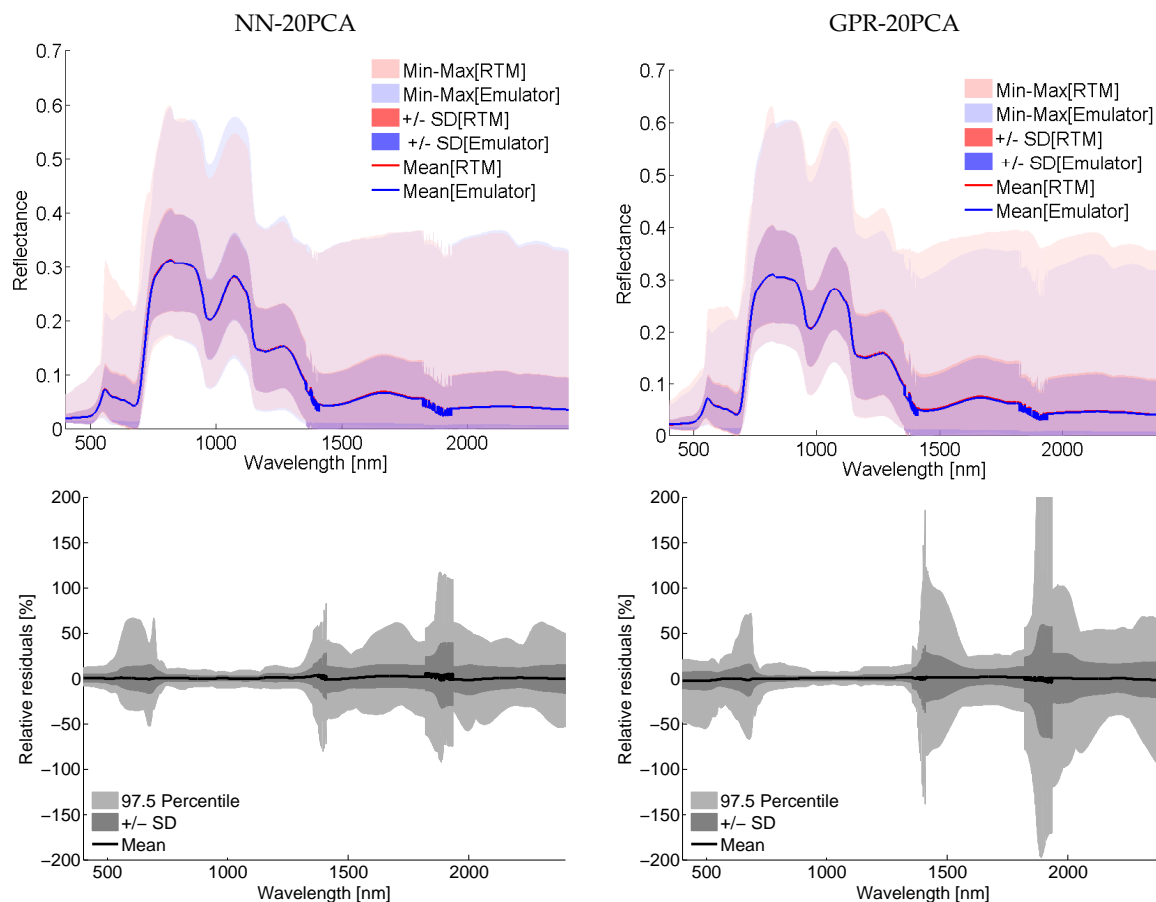
MLRA	$RMSE_{\lambda}$	$NRMSE_{\lambda}$ (%)	CPU (s)
RF	0.022	6.763	46.6
NN	0.007	2.066	266.6
SVR	0.027	8.317	810.4
KRR	0.025	7.538	2.2
GPR	0.011	3.106	114.7
VHGPR	0.008	2.433	1006.8

Next, for the most accurate MLRA method, i.e., NN, the emulation performance is inspected in more detail against SCOPE validation data. To start with, some general statistics are plotted (mean, SD min-max) of the emulator on top of the same statistics of the SCOPE validation data (Figure 2, top). From the RTM vs. emulator performance comparison it can be observed that the means are nearly perfectly overlapping, even in the 1400 and 1900 nm water absorption regions, and the same holds for the SD around the mean. Only the min-max regions show some local mismatches.

The differences in performances can probably be better inspected by plotting the general statistics of the residuals (Figure 2, bottom). The mean is centered around 0, but the SD is fluctuating and the percentiles show considerably outliers in the visible and the water absorption regions.

In comparison, the same results are also shown for GPR (Figure 2, right). GPR was chosen because of performing nearly as accurate as NN but this model was more than twice as fast trained, and considerably faster than the more advanced VHGP—although the speed in producing  $R$  spectra is what really matters. Regarding the RTM vs. emulator comparison, it can be inspected that, while again differences in the min-max plottings can be observed, the performances are similar as NN. The relative residuals display the mismatches more pronounced (Figure 2, bottom). Although extreme outliers in the visible and in the 1400 and 1900 nm water absorption regions appear, these irregular absorption

regions are typically considered as noise and discarded in imaging spectroscopy applications. Note also that the SD around the mean is broader than NN, which suggests that the GPR model is a less accurate emulator.

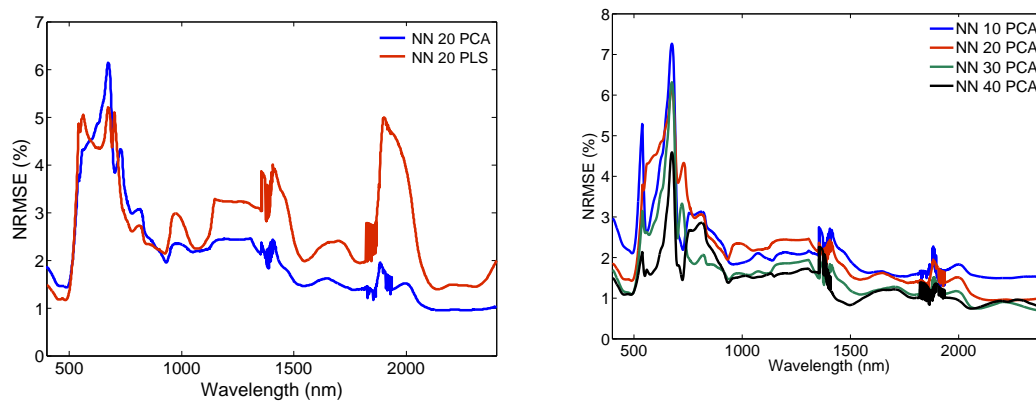


**Figure 2.** **Top:** General statistics (mean, standard deviation (SD), min-max) for SCOPE validation and corresponding emulated  $R$  spectra using SCOPE500-NN-20PCA (left) and SCOPE500-VHGPR-20PCA (right) emulators. The SD and min-max boundaries areas are presented as semi transparent so that their shapes for both datasets can be distinguished (red: RTM, blue: emulator). Consequently, the blended purple (SD) and pink (min-max) colors indicate the degree of RTM vs. emulated data overlapping. **Bottom:** Relative residuals between SCOPE and emulated  $R$  spectra. The areas represent the maximal spectral difference (light gray) and 1 SD (dark gray) for the 97.5 percentile. The black line represents the mean spectral residual.

### 5.1.2. On the Role of Dimensionality Reduction (DR) Methods

In this section, the role of dimensionality reduction (DR) methods is assessed given the best-performing MLRA method, i.e., NN. For a NN trained with 20 components the performance of PCA and PLS conversion can be inspected in Figure 3 (left). Although PLS deals slightly better with the visible part of the spectrum, on the whole, PCA delivers most accurate emulated reflectance across the spectral range. The role of number of PCA components used for training the NN models is further assessed by using 10, 20, 30 and 40 components (Figure 3, right). A clear trend appears; introducing more components into the model brings down the relative errors. Eventually, it makes that apart from the error peak in the red-edge and around 800 nm, errors are reduced to below 2% when training with 40 components. The downside of including more components is increasing the model complexity and thus a longer processing time, both for training as for applying the emulator. For instance, the training

and validation processing time increased from 66 s using 10 components to 874 s (14.5 min) using 40 components.

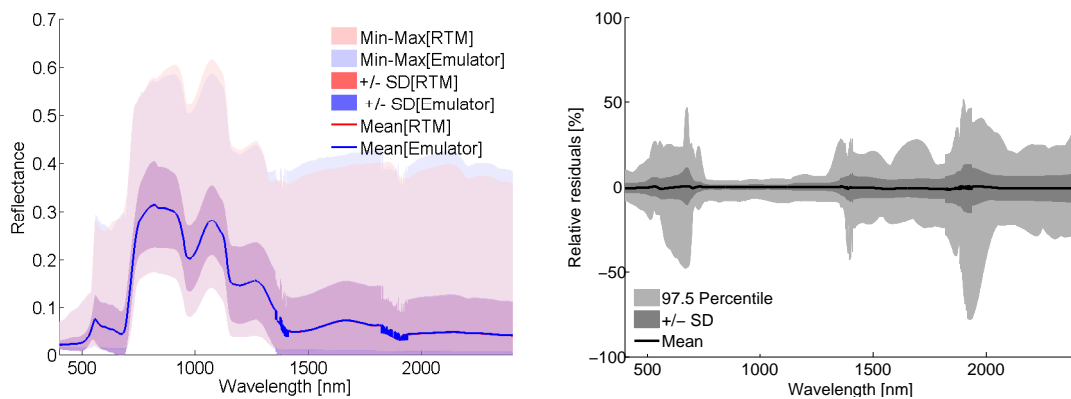


**Figure 3.** **Left:** SCOPE NRMSE [%] results for the NN validation for 30% of 500 SCOPE  $R$  spectra and using 20 components conversion for PCA and PLS. **Right:** SCOPE NRMSE [%] results for the NN validation for 30% of 500 SCOPE  $R$  spectra for PCA using 10, 20, 30 and 40 components conversion.

### 5.1.3. On the Role of LUT Size and Visualizing 500 LUT Emulator $R$ Spectra

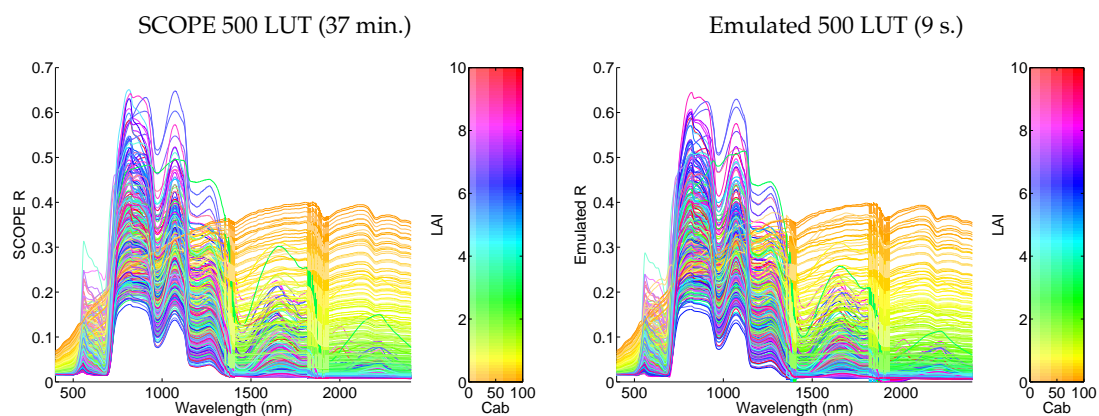
The role of LUT training size was subsequently assessed. Increasing the LUT size from 500 to 1000 samples (70% used for training) lowered the relative errors along the spectral range further (results not shown). Apart from the peak around the red-edge that lowered to 3%, relative errors are at 1% across the whole spectral range. However, the increase in LUT size goes at a computational cost. Generating the larger LUT did not only take more time (69 min), the training and validating processing of this dataset lasted even more, i.e., 85 min. Nevertheless of importance is that, once trained, applying the NN emulator delivers output very fast (e.g., 11 s to generate 500  $R$  spectra).

Since the principle of an emulator is that a small number of RTM simulations should be sufficiently representative to accurately cover the full considered parameter space (i.e., as defined by the selected input variables), we subsequently display emulated  $R$  spectra as produced by the best evaluated 500 LUT emulator, thereby keeping in mind that accuracies can still be improved when using a larger LUT size for training. Accordingly, the spectral output of the SCOPE500-NN-40PCA emulator is inspected in more detail against the associated SCOPE validation spectra. To start with, some general statistics are plotted (mean, SD min-max) of emulated  $R$  spectra on top of the same statistics coming from the SCOPE validation  $R$  spectra (Figure 4, left). It can be observed that now the mean and SD stats are perfectly overlapping, only the min-max region shows some small discrepancies. Again, the differences between SCOPE  $R$  vs. emulated  $R$  spectra are probably better expressed by the relative residuals (Figure 4, right). The mean is centered around 0 and also the SD is reduced to a few percents with some broader regions in the visible and from 1400 nm onwards. Unfortunately some outliers remain, as can be observed from the min-max plottings.



**Figure 4.** **Left:** General statistics (mean, standard deviation (SD), min-max) for SCOPE validation and corresponding emulated spectra using the SCOPE1000-NN-40PCA emulator. The SD and min-max boundaries areas are presented as semi transparent so that their shapes for both datasets can be distinguished (red: RTM, blue: emulator). Consequently, the blended purple (SD) and pink (min-max) colors indicate the degree of RTM vs. emulated data overlapping. **Right:** Relative residuals between SCOPE and emulated reflectance spectra. The areas represent the maximal spectral difference (light gray) and 1 SD (dark gray) for the 97.5 percentile. The black line represents the mean spectral residual.

To enable a visual comparison of the emulated  $R$  spectra, the 500 original LUT is firstly plotted. Although the LUT ranges over the full parameter space, the  $R$  output is color-scaled as function of LAI and Cab, (Figure 5, left). The same input values were subsequently entered into the developed emulator (NN, 40 PCA) and the LUT of the emulated  $R$  spectra is plotted next to the original LUT (Figure 5, right). Their similarity is noteworthy, despite the occurrence of some small differences. Relative errors ( $\text{NRMSE}_\lambda$ ) between these 500 SCOPE and 500 emulated  $R$  spectra is 1.7%. And this at a tremendous gain in processing speed; the 500 emulated  $R$  spectra are generated within 9 s, i.e., about 250 times faster than the SCOPE model. Obviously, the emulation was only possible after training of the 500 original simulations, but with the developed emulator LUTs of any size can be generated within the input parameter space. For instance, a LUT of ten thousand  $R$  emulations took about 8 min with the same NN emulator.

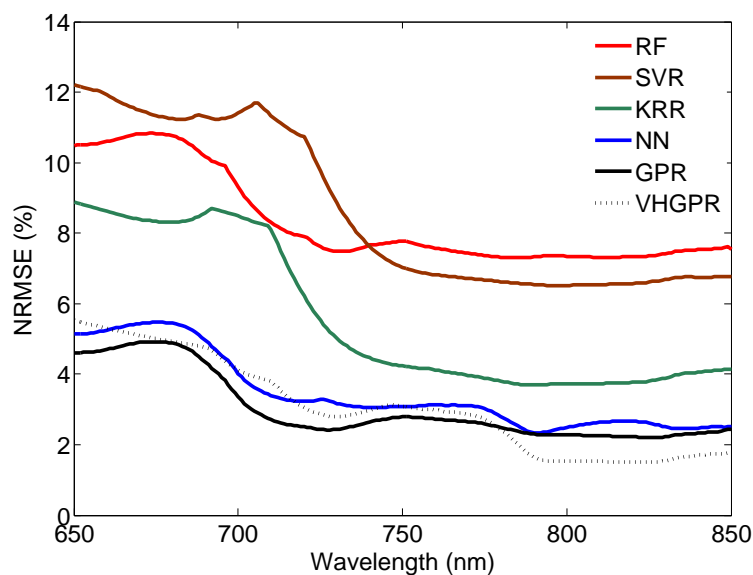


**Figure 5.** **Left:** 500 original SCOPE  $R$  spectra. **Right:** 500 emulated  $R$  spectra as generated by the SCOPE500-NN-40PCA emulator. Processing time for the generation of the 500 LUT is added on top.

## 5.2. SCOPE Sun-Induced Fluorescence (SIF) Output

### On the Role of Used MLRA

As a second part of this study, the same emulation assessment analysis was applied to emulated SCOPE SIF spectra. The main differences between SIF and R spectra is that SIF is a smooth broadband signal between 650 and 850 nm consisting of two peaks centred around 685 and 740 nm. Because of the smooth profiles, it implies that less components are needed to reconstruct the signal. Accordingly, the performance of the tested MLRAs along the spectral range are first analyzed using 5 components. Figure 6 suggests the following: (1) GPR, VHGP and NN delivered a substantially more accurate emulation than the other MLRAs; relative errors stay below 5%. GPR slightly outperforms VHGP until 790 nm, which is what matters; after 800 nm the SIF signal is close to zero. KRR delivered a substantially poorer emulator, but this method is again tremendously fast in training and emulating. Finally, the used SVR version not only performed poorest but also needed extremely long training time (Table 4).



**Figure 6.** SCOPE NRMSE [%] results for the MLRAs validation for 30% of 500 SCOPE SIF spectra using 5 PCA conversion.

**Table 4.** MLRA emulators validation results ( $RMSE_{\lambda}$ ,  $NRMSE_{\lambda}$ ) and training and validation processing time (s: seconds) for 30% of 500 SCOPE SIF outputs and using 5 PCA conversion.

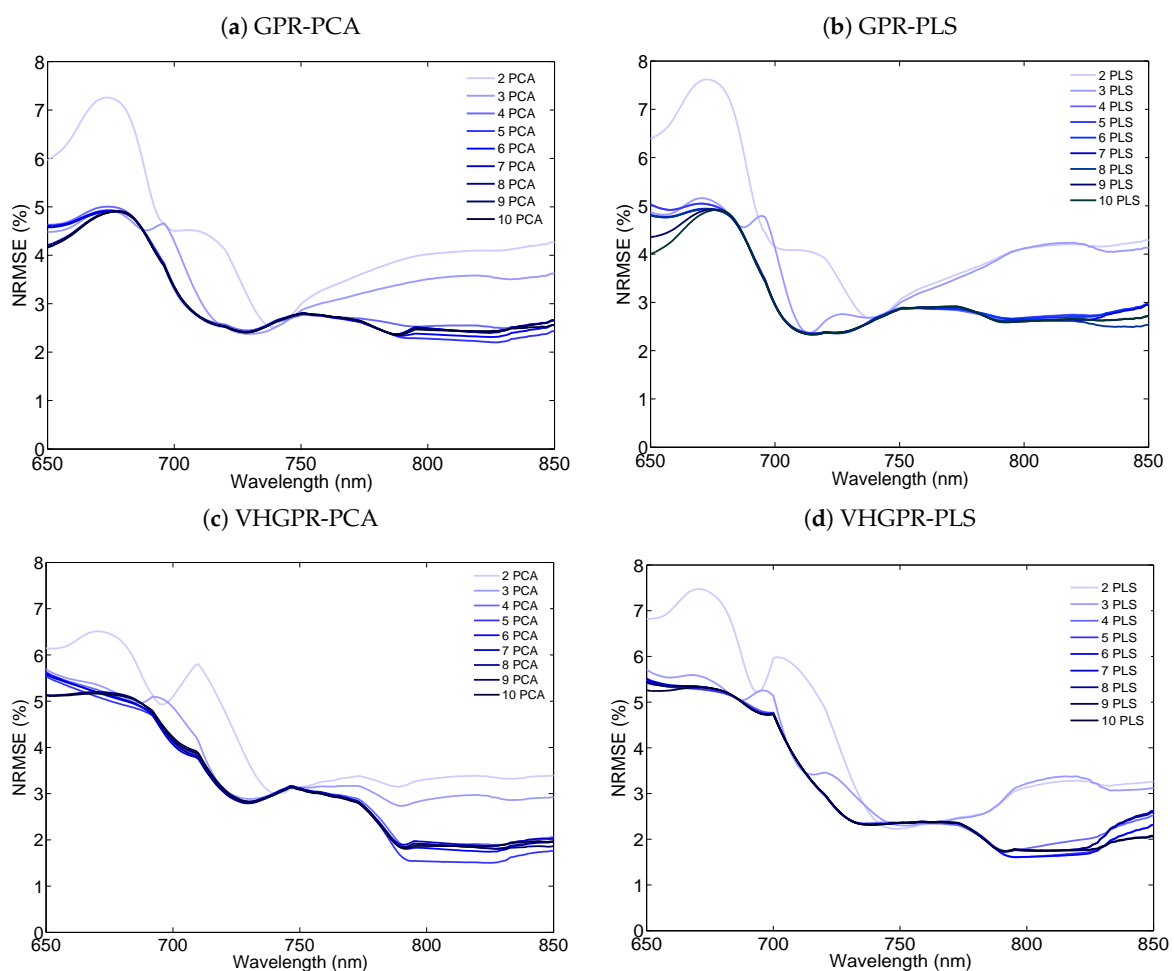
MLRA	$RMSE_{\lambda}$	$NRMSE_{\lambda}$ (%)	CPU (s)
RF	0.135	8.241	12.0
NN	0.081	5.031	44.7
SVR	0.126	7.349	293.9
KRR	0.108	6.723	1.5
GPR	0.049	3.303	27.9
VHGP	0.055	3.064	102.4

### 5.3. On the Role of Dimensionality Reduction (DR) Methods

Because GPR and VHGP yielded similar accuracies, the question arose whether these emulators can be further optimized by means of optimizing a DR method. Both PCA and PLS conversion with 2 up to 10 components were therefore evaluated. The NRMSE results of the four combinations plotted in Figure 7 suggest the following: (1) performances stabilizes after 4 components. Adding more

components hardly led to improvements while taking more processing time; (2) the differences between PCA and PLS are minimal; (3) GPR reaches higher accuracies in reproducing the SIF signal; VHGPR only reaches lower errors in the right tail. GPR runs substantially faster than VHGPR, because VHGPR adds an additional step during the training phase by exploiting the input-dependent noise. Given that both DR methods perform alike, PCA tend to be preferred because of simpler and therefore slightly faster.

It was then evaluated whether accuracies can be further improved by increasing the LUT training size to 1000 samples (70% for training) using 4 PCA components. For GPR the  $RMSE_{\lambda}$  (NRMSE $_{\lambda}$ ) lowered to 0.046 (2.957%) and for VHGPR the RMSE lowered to 0.045 (2.871%). However, when plotting the NRMSE along the spectral range then relative errors were hardly lower as compared to the 500 LUT; improvements occurred merely in the tails of the SIF signal (results not shown). Also, processing time increased more than 8 times (244 s. for GPR and 898 s. for VHGPR), which makes that a large LUT of 1000 samples is not needed to emulate SIF spectra.

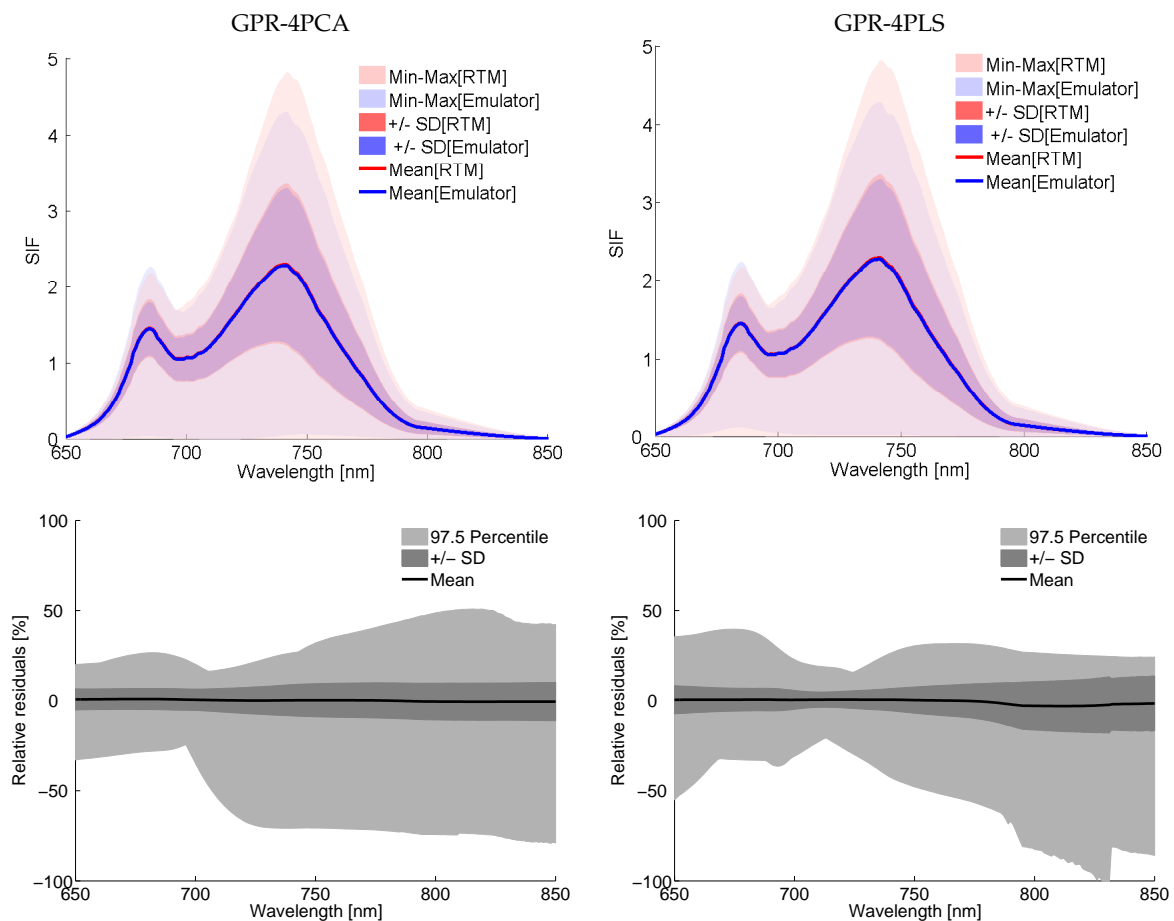


**Figure 7.** SCOPE NRMSE [%] results for the validation for 30% of 500 SCOPE SIF spectra using 2–10 (a) GPR-PCA, (b) GPR-PLS, (c) VHGPR-PCA and (d) VHGPR-PLS components conversion.

#### Visualizing 500 LUT Emulator SIF Spectra

For the two best-performing 500LUT models, GPR-4PCA and GPR-4PLS respectively, emulated SIF spectra are inspected in more detail against the original SCOPE SIF validation spectra. Again, first some general statistics (mean, SD min-max) of the emulator are plotted on top of the same statistics of the SCOPE validation spectra (Figure 8, top). From the RTM vs. emulator comparison, for both

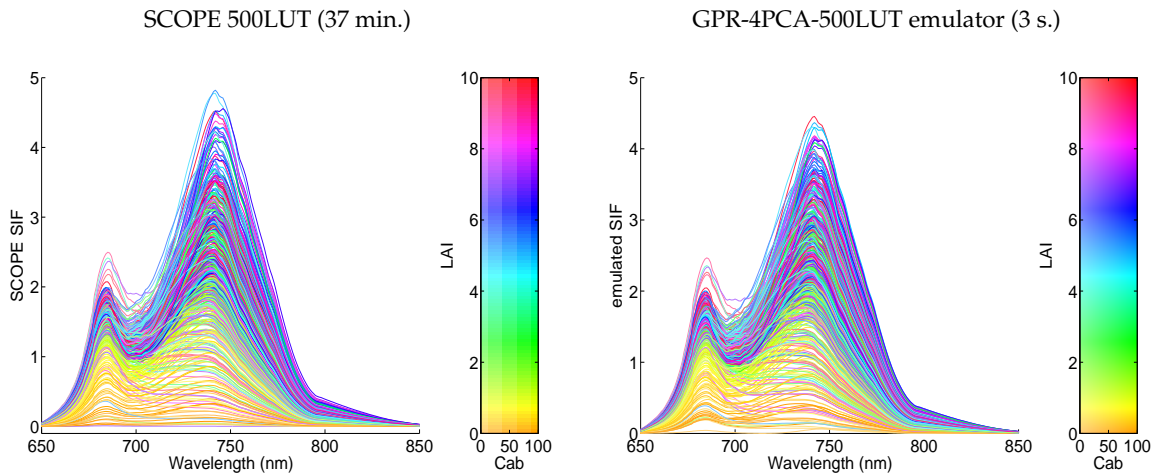
emulators the means and SD are perfectly overlapping, although there is some mismatch in the min-max regions. The differences in emulated spectra are again probably better expressed by plotting the general statistics of the residuals (Figure 8, bottom). In both cases the mean is centered around 0 although PLS shows a slight deviation around 790 nm. Also the SD of PCA stays constant throughout the spectral range, while PLS starts increasing towards the end. The percentiles are more irregular, suggesting that in both cases some outliers occurred. Altogether, PCA and PLS performed quite alike whereby PCA processes slightly faster.



**Figure 8. Top:** General statistics (mean, standard deviation (SD), min-max) for SCOPE validation and corresponding emulated *SIF* spectra using SCOPE500-GPR-4PCA (left) and SCOPE500-GPR-4PLS (right) emulators. The SD and min-max boundaries areas are presented as semi transparent so that their shapes for both datasets can be distinguished (red: RTM, blue: emulator). Consequently, the blended purple (SD) and pink (min-max) colors indicate the degree of RTM vs. emulated data overlapping. **Bottom:** Relative residuals between SCOPE and emulated *SIF* spectra. The areas represent the maximal spectral difference (light gray) and 1 standard deviation (dark gray) for the 97.5 percentile. The black line represents the mean spectral residual.

Finally, just as with the *R* spectra, the original 500 SCOPE *SIF* spectra are plotted next to the same *SIF* spectra as produced by the GPR-4PCA emulator (Figure 9). The emulated spectra are delivered in 3 s as opposed to 37 min by SCOPE. Although the full parameter space of the 8 input variables is covered, the plotting is again color-scaled according to LAI and Cab. The same *SIF* spectra are produced but some small differences may appear for some emulations. Relative errors (NRMSE<sub>λ</sub>) between these 500 SCOPE and 500 emulated *SIF* spectra is 2.1%.

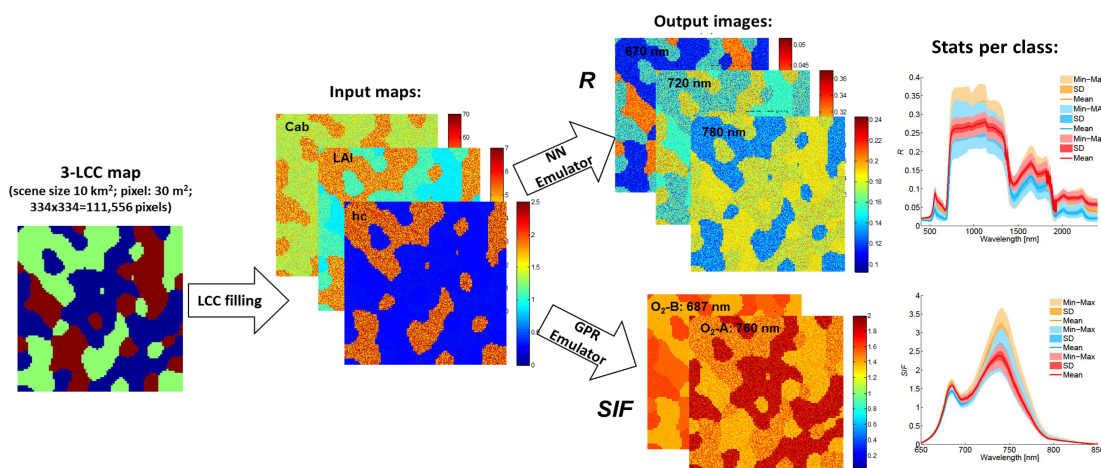




**Figure 9.** Left: 500 original SCOPE SIF spectra. Right: 500 emulated SIF spectra as generated by the SCOPE500 GPR-4PCA emulator. Processing time for the generation of the 500 LUT is added on top.

5.4. Scene Generation

As a demonstration case, we close this work with the rendering of hyperspectral scenes of *R* and *SIF* spectra as generated by emulators (Figure 10). To do so, the best-performing 500 LUT emulator for *R* (i.e., SCOPE500-NN-40PCA) and the close to best-performing but fast emulator for *SIF* (i.e., SCOPE500-GPR-4PCA) were imported into the A-SGM to generate *R* and *SIF* spectra of a vegetated scene defined according to the vegetation classes of Table 2. Once having the scene properties and vegetation classes into the A-SGM defined, the toolbox produces both maps of the input variables and renders an output scene. Here a scene consisting of 111,556 hyperspectral pixels was emulated. Although a GPR emulator is generally slower than a NN emulator, because of having only 4 components involved as opposed to 40, the *SIF* scene was generated twice as fast as the *R* scene; the rendering of the *R* scene took about 24 min (NN emulator) whereas the *SIF* scene took 11 min (GPR emulator). The A-SGM further provides the option to analyze the generated scene by some simple stats (e.g., plotting output statistics per class), as exemplified in Figure 10 (right).



**Figure 10.** Illustration of A-SGM scene generation by using emulators. First a land cover map is arbitrarily created. Then these classes are filled with the input variables according to Table 2. Following the emulators are run to generate the spectral datacubes. For illustration, output images of a few wavelengths are shown. Also general statistics (mean, SD, min-max) per class are derived.

## 6. Discussion

Let us now discuss on the most important messages conveyed in this work, and the implications on further studies for imaging spectroscopy data analysis, scene generation and model inversion.

### 6.1. Interpreting Emulator Results

Emulation is a statistical technique used to approximate a deterministic model when the model is too computationally expensive to be run many times [23]. Following on initial successful experiments that MLRAs can function as emulators to approximate RTMs [24–26], here the concept of emulating an advanced RTM has been further explored. SCOPE was the RTM under study, but the emulation technique can be applied to any RTM. Specifically, key aspects that determine the degree of accuracy that can be reached through emulators were addressed. These aspects include: (1) used machine learning algorithm, (2) used dimensionality reduction method, and (3) number of components. Each of these aspects are briefly discussed below.

From the 6 machine learning algorithms tested, only those of the family of Gaussian processes regression (GPR and the variational heteroschedastic GPR variant) and neural networks (NN) reached best accuracies. The other popular machine learning algorithms such as support vector regression (SVR) and random forests (RF) were in none of the tested experiments able to reach the accuracy as delivered by GPR or NN. Kernel ridge regression (KRR) performed in-between those two groups and deserves special attention because of being trained and running extremely fast. For instance, by using the KRR model as trained by 40 components it took about 1 s to generate 500 *R* spectra. Hence, for situations where accuracy is less of an issue, e.g., during algorithm development and testing phase, then KRR can serve as an attractive emulator for rapid spectral data generation. KRR is faster because it has less hyperparameters to tune. GPR and VHGP, however, are not multi-output methods but instead loop over the components. Consequently, the training and application of these models will process slower when more components are involved. But the family of GPR has an additional advantage that these Bayesian methods provide associated uncertainty estimates [69]. Such information can function as a filter, i.e., discard emulations when surpassing an uncertainty threshold. Finally, although the best-performing emulators reached accuracies with relative errors (NRMSE) of 2 to 4% for the tested datasets, improvements may still be possible when exploiting latest generation of machine learning algorithms, e.g., in deep learning [70,71].

A second aspect that was evaluated was whether the dimensionality reduction (DR) method PLS as an alternative of the classical PCA would lead to higher accuracies. The DR step is of importance in the emulation of RTM spectral outputs, as it allows compressing the spectral output into a manageable number of components before entering into the regression algorithm, and afterwards reconstructing the full spectrum again. This reconstruction step is mandatory to be able delivering spectroscopic data outputs with several hundreds to thousands bands. Even though PLS is a slightly more advanced version of PCA, because both methods are mathematically alike, PLS hardly led to notable improvements. While for *R* emulation PLS led to higher accuracies in some spectral regions, it generally performed poorer and more unstable than PCA; and for *SIF* emulation both DR methods performed alike. Altogether, PCA can be concluded as the preferred DR method, though the development of powerful DR methods with capabilities to reconstruct the signal is strongly encouraged.

A third aspect determining the emulator accuracy involves the number of components into the regression model. To some extent, including more components improves the emulation accuracy. By adding more components to the regression model in principle more spectral variability is preserved. But at some point saturation takes place, and adding more components instead decelerates the emulation process. However, the optimal number of components is rather driven by the complexity of spectral data, i.e., more irregular spectral data require more components to preserve the full spectral variability. This was observed for the *R* emulation where a progressive increase in accuracy was reached by adding more components. Typically, eventually accuracy stability will be reached,

although that was not yet the case with 40 components. Stability was more rapidly achieved with the spectrally smooth 2-peak *SIF* emulation. After 4 components improvements were minimal to non-existent. Conversely, when moving towards more irregular top-of-atmosphere (TOA) radiance data [25] then it appeared considerably more components are required to reach accurate emulation results. These examples underline that the nature of the spectral data determines the optimized number of components. Practically, some repetitions are required to deduce this optimal number.

A fourth aspect is that a larger LUT, i.e. more samples used for training, can also favor the emulator accuracy. The LUT size is worth exploring when aiming to maximize emulation precision. On the other hand, it should be remarked that a larger LUT goes together with a longer processing time, both in LUT generation, model training and model running. Especially NN needs a long training time, yet once the model is developed, it generates spectral outputs very fast. GPR is faster in training but takes more time to produce output when many components are involved.

A final point to be remarked is that the emulator performance also depends on the number and nature of included input variables. Adding more input variables amplifies the complexity of the statistical model and may thus be at the expense of accuracy and processing speed. However, not all input variables play an equally important role in shaping spectral output and over the complete spectral range. In case of SCOPE, that provides all kinds of outputs, some input variables have no influence at all to some outputs, e.g., soil variables in case of *SIF* output [20]. This means that in principle only the driving input variables related to a targeted output are of interest to develop an emulator; including uninfluential variables would only make the emulator unnecessary heavy. A global sensitivity analysis can infer the wavelength-dependent driving input variables, but also some input variables can be kept fixed because of typically known, e.g., sun-target-sensor geometry and perhaps some meteorological variables. This means that the development of emulators can be customized depending on available information about input variables, the targeted spectral output and the intended application.

## 6.2. Opportunities for Fast Scene Generation

As a case study, the applicability of emulators has been demonstrated by importing them into ARTMO's scene generator module (A-SGM). The generation of a scene consisting of 111,556 pixels took about 24 min for a *R* datacube and 11 min for a *SIF* datacube. In comparison, using SCOPE would take more than 5 days to generate the same scene. At the same time, it did not escape our attention that the rendered *R* and *SIF* scenes are of low realism in representing vegetated areas. While here shown as proof of concept only, in principle the A-SGM can render realistic vegetation classes when input variables ranges are supported by field data and the spatial distribution is defined according to a land cover map. Moreover, when entering temporal profiles of the input variables then also time series of multiple images can be generated. Hence, when using real land cover maps and real time series of input variables, then the A-SGM can become an attractive tool to produce more realistic scenes.

A drawback of the current version, however, is that the A-SGM is limited to generate TOC spectral outputs only. In order to upscale to TOA radiance data, a coupling with an atmospheric model is required. Such coupling is already implemented in the FLEX E2E simulator (FLEX-E) through the atmospheric RTM MODTRAN [13]. Yet, adding an additional RTM to the system will further slow down the processing chain. In fact, MODTRAN is even more computationally expensive than SCOPE and is currently the main bottleneck of FLEX-E. To enable fast generation of TOA radiance data, it would therefore make a substantial difference to have MODTRAN emulated and couple it with a SCOPE-like emulator. Progress in this direction is underway [72]. Based on these developments, it is expected that ultimately E2E Simulators such as FLEX-E will have the option to run multiple emulators in order to generate quasi-simultaneously synthetic TOC and TOA scenes.

### 6.3. New Processing Opportunities with Emulators

The most significant advantage of customized RTM-like emulators is the tremendous gain in processing speed—although at a risk of losing somewhat in precision as compared to the original RTM (here reaching relative errors down to 2% when comparing original vs. emulated *R* and *SIF* LUTs). The tested emulators deliver spectral outputs quasi-instantly, which implies accelerating processing speed in the orders of hundreds to thousands times, depending on the speed of the original RTM. At the same time, they hardly take memory space, only a few model coefficients are stored for prediction. Consequently, emulators can become an attractive technique for a diversity of remote sensing applications when accepting a small loss in output precision. A few application examples have already been mentioned in [25], here they are further elaborated and some more are added.

*Global sensitivity analysis.* An important aspect when using advanced RTMs is the identification of the key input variables driving the spectral output and variables of lesser influence. By identifying variables of lesser influence, models can be greatly simplified, which facilitates practical applications such as inversion [73]. To achieve this, a sensitivity analysis is required. Global sensitivity analysis (GSA) explores the full input variable space [74]. Specifically, variance-based GSA methods aim to quantify the amount of variance that each input variable contributes to the unconditional variance (variance across all simulations) of the model's output [75]. However, a GSA requires many simulations (depending on the number of variables, typically in the orders of ten thousands), which makes it computationally demanding. As an elegant alternative, recently it has been demonstrated that GSA calculation can go very fast when replacing the RTM by an emulator [25]. Multiple emulators of leaf, canopy and atmosphere RTMs were analyzed, including MODTRAN. In this way, to the best of our knowledge, for the first time MODTRAN driving variables on the atmospheric transfer functions were identified. The processing of GSA took only a few minutes using an emulator – in comparison it would take over a month using MODTRAN.

*Numerical inversion.* Iterative optimization is a classical technique to invert RTMs against spectral data [76,77]. The optimization consists in minimizing a cost function, which estimates the difference between measured and estimated variables by successive input variable iteration. The iterative optimization algorithms are computationally demanding and hence time-consuming when large spectral datasets are inverted. This method has never been a viable option for computationally intensive RTMs, and even for computationally cheap RTMs the method is slow and not appealing. Making use of pre-generated LUTs is preferred in inversion schemes (see [48] for a review). For instance, canopy RTM LUTs are used in the MODIS LAI retrieval products [78] and also atmospheric correction algorithms make use of LUT-based inversion methods e.g., [79–83]. Accordingly, when replacing the original RTMs by their emulated counterparts then numerical inversion can again become an attractive alternative; numerical inversion using a RTM-like emulator can go very fast. It would bypass the need to invest in large and heavy LUTs. To explore this research line further, in ARTMO a so-called Numerical Inversion Toolbox is currently under construction that will make use of emulators. However, since a scene typically consists of millions of pixels and per pixel many iterations are required, initial tests suggested that only the KRR emulator is fast enough to deliver output maps within reasonable time.

*LUT generation and exploiting.* Alternatively, emulation can also be a viable option for large LUT generation. Although a small LUT is required to develop an emulator, the emulator can subsequently be used to generate a large LUT—that is essentially how scene generation or GSA works. Yet another possibility of emulation is to function as an alternative of interpolation techniques in LUT-based inversion strategies. This can become an attractive approach, since interpolation techniques are widely used to fill up the parameter space when developing very large LUTs are too computationally expensive e.g., [82–85]. A follow-up study will infer whether emulation techniques would lead to more accurate and faster gap-filling predictions as is done by conventional interpolation techniques.

*Emulation of experimental data.* Beyond RTMs, the emulation technique can likewise be applied to experimental data. Instead of being trained by synthetic data, just as well field data (i.e.,

variables plus associated spectral data) can be used to train a MLRA. As such, an emulator is developed with the capability of generating spectral outputs similar to what is observed in the field. This gap-filling technique can be of interest, for instance when insufficient field data has been collected. Another advantage of building such an emulator is that the driving input variables in shaping the spectral data can be rapidly identified by running a GSA to the emulator. In addition, the emulator can also be implemented into inversion schemes. These various research paths will be further explored in follow-up studies.

## 7. Conclusions

Emulators are statistical constructs that approximate the functioning of deterministic models such as RTMs. If proven sufficiently accurate, they provide great savings in memory and tremendous gains in processing speed, and open many new research and practical opportunities—not in the least enabling to convert computationally expensive RTMs into computationally fast surrogate models. Given a subset of input variables, we analyzed key aspects of emulator development that play a role in the accuracy of reproducing SCOPE canopy-leaving reflectance ( $R$ ) and fluorescence ( $SIF$ ) spectra, being: the used (1) machine learning regression algorithm, (2) dimensionality reduction (DR) method to transform the spectral dimension into components, and (3) number of components involved into the regression algorithm. The machine learning family of Gaussian processes regression (GPR) and neural networks (NN) were found most powerful to function as emulator, whereas kernel ridge regression (KRR) reconstructs spectral outputs extremely fast at the expense of losing somewhat in accuracy. PCA remains a robust DR method in decomposing and reconstructing the spectra, but the number of components needs to be optimized depending on the complexity of the spectral data to preserve maximal spectral variability. Based on a small LUT of 500 samples (70% used for training), the best-performing emulators can reconstruct any combination of the given SCOPE input variables; they were validated with relative errors below 2% for  $R$  and 4% for  $SIF$ , and errors can still go down when using a larger LUT size. As a proof of concept, the best-performing  $R$  and  $SIF$  emulators were subsequently imported into ARTMO's new Scene Generator Module (A-SGM) to produce synthetic  $R$  and  $SIF$  scenes of a vegetated surface. It did not take longer than 24 and 11 min, respectively, to create hyperspectral datacubes of  $R$  (by NN emulator) and  $SIF$  (by GPR emulator) with a size of  $334 \times 334$  pixels. In comparison, SCOPE would need multiple days to produce the same scenes. With ongoing progress made in machine learning, it is expected that in the forthcoming years emulators will be able to produce spectral output that is fully consistent with the original RTM. As such, emulators can replace computationally expensive physical models in all kinds of data processing applications, not only in the field of radiative transfer modelling for spectroscopic data analysis, but in any field where modeling is involved.

**Acknowledgments:** This work was done as part of the AVANFLEX (Advanced Products for the FLEX mission) project, ESP2016-79503-C2-1-P, and supported by the Spanish Ministry of Economy and Competitiveness through the National Program for the Promotion of Scientific and Technical Research of Excellence. Jordi Muñoz-Marí wants to thank the Spanish MINECO and European ERDF funding under project TIN2015-64210-R. Gustau Camps-Valls wants to thank the European Research Council (ERC) funding under the ERC-CoG-2014 SEDAL under grant agreement 647423. We thank the three reviewers for their valuable suggestions.

**Author Contributions:** Jochem Verrelst performed the calculus and wrote the paper, Juan Pablo Rivera Caicedo and Jordi Muñoz-Marí developed the emulator toolbox and helped with the emulator theory and results generation. Gustau Camps-Valls contributed to the paper structure and improved the paper English style and José Moreno supervised the full study.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Appendix A

Table A1. SCOPE v. 1.61 fixed input parameters according to default settings.

Broadband Incoming Shortwave Radiation (0.4–2.5 $\mu\text{m}$ ) [ $\text{W m}^{-2}$ ]	600
air temperature [T]	20
air pressure [hPa]	970
atmospheric vapour pressure [hPa]	15
wind speed at height z [ $\text{m s}^{-1}$ ]	2
atmospheric $\text{CO}_2$ concentration [ppm]	380
atmospheric $\text{O}_2$ concentration [per mille]	209
carotenoid content [ $\mu\text{g cm}^{-2}$ ]	20
leaf width [m]	0.1
soil boundary layer resistance [ $\text{s m}^{-1}$ ]	10
within canopy layer resistance [ $\text{s m}^{-1}$ ]	0
leaf boundary resistance [ $\text{s m}^{-1}$ ]	10
observation zenith angle [ $^\circ$ ]	0
solar zenith angle [ $^\circ$ ]	30
azimuthal difference between solar and observation angle [ $^\circ$ ]	0
soil resistance for evaporation from the pore space [ $\text{s m}^{-1}$ ]	500
spectrum number (column in the database soil file) []	1
maximum carboxylation capacity (at optimum temperature) [ $\mu\text{mol m}^{-2} \text{s}^{-1}$ ]	30
Ball-Berry stomatal conductance parameter []	8
extinction coefficient for $V_{\text{cmax}}$ in the vertical (maximum at the top). 0 for uniform $V_{\text{cmax}}$ []	0.6396
Respiration = $\text{Rdparam} * V_{\text{cmax}}$ []	0.0150
fluorescence model	Collatz-TB12
measurement height of meteorological data-[m]	10
broadband thermal transmittance	0.01
photochemical pathway	C3
fluorescence quantum yield efficiency at photosystem level	0.02
broadband soil reflectance in the thermal range (1-emissivity)	0.06
leaf drag coefficient	0.3
leaf inclination distribution	spherical

## References

- Schaepman, M.E.; Ustin, S.L.; Plaza, A.J.; Painter, T.H.; Verrelst, J.; Liang, S. Earth system science related imaging spectroscopy—An assessment. *Remote Sens. Environ.* **2009**, *113*, S123–S137.
- Li, L.; Zhang, Q.; Huang, D. A review of imaging techniques for plant phenotyping. *Sensors* **2014**, *14*, 20078–20111.
- Asner, G.; Martin, R. Spectranomics: Emerging science and conservation opportunities at the interface of biodiversity and remote sensing. *Glob. Ecol. Conserv.* **2016**, *8*, 212–219.
- Rossini, M.; Meroni, M.; Migliavacca, M.; Manca, G.; Cogliati, S.; Busetto, L.; Picchi, V.; Cescatti, A.; Seufert, G.; Colombo, R. High resolution field spectroscopy measurements for estimating gross ecosystem production in a rice field. *Agric. For. Meteorol.* **2010**, *150*, 1283–1296.
- Daumard, F.; Champagne, S.; Fournier, A.; Goulas, Y.; Ounis, A.; Hanocq, J.F.; Moya, I. A field platform for continuous measurement of canopy fluorescence. *IEEE Trans. Geosci. Remote Sens.* **2010**, *48*, 3358–3368.
- Rascher, U.; Alonso, L.; Burkart, A.; Cilia, C.; Cogliati, S.; Colombo, R.; Damm, A.; Drusch, M.; Guanter, L.; Hanus, J.; et al. Sun-induced fluorescence—A new probe of photosynthesis: First maps from the imaging spectrometer HyPlant. *Glob. Chang. Biol.* **2015**, *21*, 4673–4684.
- Drusch, M.; Moreno, J.; Del Bello, U.; Franco, R.; Goulas, Y.; Huth, A.; Kraft, S.; Middleton, E.M.; Miglietta, F.; Mohammed, G.; et al. The FLuorescence EXplorer mission concept-ESA's Earth Explorer 8. *IEEE Trans. Geosci. Remote Sens.* **2016**, *55*, 1273–1284.
- Ball, J.T.; Woodrow, I.E.; Berry, J.A. *A Model Predicting Stomatal Conductance and Its Contribution to the Control of Photosynthesis under Different Environmental Conditions*; Springer: Dordrecht, The Netherlands, 1987; pp. 221–224.

9. Collatz, G.J.; Ball, J.T.; Grivet, C.; Berry, J.A. Physiological and environmental regulation of stomatal conductance, photosynthesis and transpiration: A model that includes a laminar boundary layer. *Agric. For. Meteorol.* **1991**, *54*, 107–136.
10. Collatz, G.J.; Ribas-Carbo, M.; Berry, J. Coupled photosynthesis-stomatal conductance model for leaves of C4 plants. *Funct. Plant Biol.* **1992**, *19*, 519–538.
11. Gausman, H.; Allen, W.; Cardenas, R.; Richardson, A. Relation of light reflectance to histological and physical evaluations of cotton leaf maturity. *Appl. Opt.* **1970**, *9*, 545–552.
12. Jacquemoud, S.; Baret, F. PROSPECT: A model of leaf optical properties spectra. *Remote Sens. Environ.* **1990**, *34*, 75–91.
13. Berk, A.; Anderson, G.; Acharya, P.; Bernstein, L.; Muratov, L.; Lee, J.; Fox, M.; Adler-Golden, S.; Chetwynd, J.; Hoke, M.; et al. MODTRAN<sup>TM</sup>5: 2006 Update. *SPIE* **2006**, 6233, doi:10.1117/12.665077.
14. Jacquemoud, S.; Verhoef, W.; Baret, F.; Bacour, C.; Zarco-Tejada, P.; Asner, G.; François, C.; Ustin, S. PROSPECT + SAIL models: A review of use for vegetation characterization. *Remote Sens. Environ.* **2009**, *113*, S56–S66.
15. Pedrós, R.; Goulas, Y.; Jacquemoud, S.; Louis, J.; Moya, I. FluorMODleaf: A new leaf fluorescence emission model based on the PROSPECT model. *Remote Sens. Environ.* **2010**, *114*, 155–167.
16. Vilfan, N.; van der Tol, C.; Muller, O.; Rascher, U.; Verhoef, W. Fluspect-B: A model for leaf fluorescence, reflectance and transmittance spectra. *Remote Sens. Environ.* **2016**, *186*, 596–615.
17. Van der Tol, C.; Berry, J.A.; Campbell, P.K.E.; Rascher, U. Models of fluorescence and photosynthesis for interpreting measurements of solar-induced chlorophyll fluorescence. *J. Geophys. Res. Biogeosci.* **2014**, *119*, 2312–2327.
18. Gastellu-Etchegorry, J.; Lauret, N.; Yin, T.; Landier, L.; Kallel, A.; Malenovský, Z.; Al Bitar, A.; Aval, J.; Benhmida, S.; Qi, J.; et al. DART: Recent advances in remote sensing data modeling with atmosphere, polarization, and chlorophyll fluorescence. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2017**, *10*, 2640–2649.
19. Hernández-Clemente, R.; North, P.; Hornero, A.; Zarco-Tejada, P. Assessing the effects of forest health on sun-induced chlorophyll fluorescence using the FluorFLIGHT 3-D radiative transfer model to account for forest structure. *Remote Sens. Environ.* **2017**, *193*, 165–179.
20. Verrelst, J.; Rivera, J.P.; van der Tol, C.; Magnani, F.; Mohammed, G.; Moreno, J. Global sensitivity analysis of the SCOPE model: What drives simulated canopy-leaving sun-induced fluorescence? *Remote Sens. Environ.* **2015**, *166*, 8–21.
21. Verrelst, J.; van der Tol, C.; Magnani, F.; Sabater, N.; Rivera, J.; Mohammed, G.; Moreno, J. Evaluating the predictive power of sun-induced chlorophyll fluorescence to estimate net photosynthesis of vegetation canopies: A SCOPE modeling study. *Remote Sens. Environ.* **2016**, *176*, 139–151.
22. Kennedy, M.; O'Hagan, A. Predicting the output from a complex computer code when fast approximations are available. *Biometrika* **2000**, *87*, 1–13.
23. O'Hagan, A. Bayesian analysis of computer code outputs: A tutorial. *Reliab. Eng. Syst. Saf.* **2006**, *91*, 1290–1300.
24. Gómez-Dans, J.L.; Lewis, P.E.; Disney, M. Efficient emulation of radiative transfer codes using Gaussian Processes and application to land surface parameter inferences. *Remote Sens.* **2016**, *8*, 119.
25. Verrelst, J.; Sabater, N.; Rivera, J.P.; Muñoz Marí, J.; Vicent, J.; Camps-Valls, G.; Moreno, J. Emulation of leaf, canopy and atmosphere radiative transfer models for fast global sensitivity analysis. *Remote Sens.* **2016**, *8*, 673.
26. Rivera, J.P.; Verrelst, J.; Gómez-Dans, J.; Muñoz Marí, J.; Moreno, J.; Camps-Valls, G. An emulator toolbox to approximate radiative transfer models with statistical learning. *Remote Sens.* **2015**, *7*, 9347.
27. Verrelst, J.; Romijn, E.; Kooistra, L. Mapping vegetation density in a heterogeneous river floodplain ecosystem using pointable CHRIS/PROBA data. *Remote Sens.* **2012**, *4*, 2866–2889.
28. Vicent, J.; Sabater, N.; Tenjo, C.; Acarreta, J.; Manzano, M.; Rivera, J.; Jurado, P.; Franco, R.; Alonso, L.; Verrelst, J.; et al. FLEX End-to-End mission performance simulator. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 4215–4223.
29. Cogliati, S.; Verhoef, W.; Kraft, S.; Sabater, N.; Alonso, L.; Vicent, J.; Moreno, J.; Drusch, M.; Colombo, R. Retrieval of sun-induced fluorescence using advanced spectral fitting methods. *Remote Sens. Environ.* **2015**, *169*, 344–357.

30. Damm, A.; Guanter, L.; Paul-Limoges, E.; van der Tol, C.; Hueni, A.; Buchmann, N.; Eugster, W.; Ammann, C.; Schaepman, M. Far-red sun-induced chlorophyll fluorescence shows ecosystem-specific relationships to gross primary production: An assessment based on observational and modeling approaches. *Remote Sens. Environ.* **2015**, *166*, 91–105.
31. Van der Tol, C.; Rossini, M.; Cogliati, S.; Verhoef, W.; Colombo, R.; Rascher, U.; Mohammed, G. A model and measurement comparison of diurnal cycles of sun-induced chlorophyll fluorescence of crops. *Remote Sens. Environ.* **2016**, *186*, 663–677.
32. Zhang, Y.; Guanter, L.; Berry, J.; van der Tol, C.; Yang, X.; Tang, J.; Zhang, F. Model-based analysis of the relationship between sun-induced chlorophyll fluorescence and gross primary production for remote sensing applications. *Remote Sens. Environ.* **2016**, *187*, 145–155.
33. Petropoulos, G.; Wooster, M.; Carlson, T.; Kennedy, M.; Scholze, M. A global Bayesian sensitivity analysis of the 1D SimSphere soil vegetation atmospheric transfer (SVAT) model using Gaussian model emulation. *Ecol. Model.* **2009**, *220*, 2427–2440.
34. Rohmer, J.; Foerster, E. Global sensitivity analysis of large-scale numerical landslide models based on Gaussian-Process meta-modeling. *Comput. Geosci.* **2011**, *37*, 917–927.
35. Carnevale, C.; Finzi, G.; Guariso, G.; Pisoni, E.; Volta, M. Surrogate models to compute optimal air quality planning policies at a regional scale. *Environ. Model. Softw.* **2012**, *34*, 44–50.
36. Villa-Vialaneix, N.; Follador, M.; Ratto, M.; Leip, A. A comparison of eight metamodeling techniques for the simulation of N<sub>2</sub>O fluxes and N leaching from corn crops. *Environ. Model. Softw.* **2012**, *34*, 51–66.
37. Castelletti, A.; Galelli, S.; Ratto, M.; Soncini-Sessa, R.; Young, P. A general framework for dynamic emulation modelling in environmental problems. *Environ. Model. Softw.* **2012**, *34*, 5–18.
38. Lee, L.; Pringle, K.; Reddington, C.; Mann, G.; Stier, P.; Spracklen, D.; Pierce, J.; Carslaw, K. The magnitude and causes of uncertainty in global model simulations of cloud condensation nuclei. *Atmos. Chem. Phys.* **2013**, *13*, 8879–8914.
39. Bounceur, N.; Crucifix, M.; Wilkinson, R. Global sensitivity analysis of the climate—Vegetation system to astronomical forcing: An emulator-based approach. *Earth Syst. Dyn. Discuss.* **2014**, *5*, 901–943.
40. Ireland, G.; Petropoulos, G.; Carlson, T.; Purdy, S. Addressing the ability of a land biosphere model to predict key biophysical vegetation characterisation parameters with Global Sensitivity Analysis. *Environ. Model. Softw.* **2015**, *65*, 94–107.
41. Camps-Valls, G.; Bruzzone, L. (Eds.) *Kernel Methods for Remote Sensing Data Analysis*; Wiley & Sons: Chichester, UK, 2009.
42. Razavi, S.; Tolson, B.A.; Burn, D.H. Numerical assessment of metamodeling strategies in computationally intensive optimization. *Environ. Model. Softw.* **2012**, *34*, 67–86.
43. McKay, M.; Beckman, R.; Conover, W. Comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics* **1979**, *21*, 239–245.
44. Baret, F.; Clevers, J.; Steven, M. The robustness of canopy gap fraction estimates from red and near-infrared reflectances: A comparison of approaches. *Remote Sens. Environ.* **1995**, *54*, 141–151.
45. Combal, B.; Baret, F.; Weiss, M.; Trubuil, A.; Macé, D.; Pragnère, A.; Myneni, R.; Knyazikhin, Y.; Wang, L. Retrieval of canopy biophysical variables from bidirectional reflectance using prior information to solve the ill-posed inverse problem. *Remote Sens. Environ.* **2003**, *84*, 1–15.
46. Verrelst, J.; Muñoz, J.; Alonso, L.; Delegido, J.; Rivera, J.; Camps-Valls, G.; Moreno, J. Machine learning regression algorithms for biophysical parameter retrieval: Opportunities for Sentinel-2 and -3. *Remote Sens. Environ.* **2012**, *118*, 127–139.
47. Rivera, J.; Verrelst, J.; Delegido, J.; Veroustraete, F.; Moreno, J. On the semi-automatic retrieval of biophysical parameters based on spectral index optimization. *Remote Sens.* **2014**, *6*, 4924–4951.
48. Verrelst, J.; Camps Valls, G.; Muñoz Mari, J.; Rivera, J.; Veroustraete, F.; Clevers, J.; Moreno, J. Optical remote sensing and the retrieval of terrestrial vegetation bio-geophysical properties—A review. *ISPRS J. Photogramm. Remote Sens.* **2015**, *108*, 273–290.
49. Hankin, R.K. Introducing BACCO, an R package for Bayesian analysis of computer code output. *J. Stat. Softw.* **2005**, *14*, 1–21.
50. Hughes, G. On the mean accuracy of statistical pattern recognizers. *IEEE Trans. Inf. Theory* **1968**, *14*, 55–63.
51. Wold, S.; Esbensen, K.; Geladi, P. Principal component analysis. *Chemom. Intell. Lab. Syst.* **1987**, *2*, 37–52.
52. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32.



53. Haykin, S. *Neural Networks—A Comprehensive Foundation*, 2nd ed.; Prentice Hall: Upper Saddle River, NJ, USA, 1999.
54. Shawe-Taylor, J.; Cristianini, N. *Kernel Methods for Pattern Analysis*; Cambridge University Press: New York, NY, USA, 2004.
55. Vapnik, V.; Golowich, S.; Smola, A. Support vector method for function approximation, regression estimation, and signal processing. *Adv. Neural Inf. Process. Syst.* **1997**, *9*, 281–287.
56. Suykens, J.; Vandewalle, J. Least squares support vector machine classifiers. *Neural Process. Lett.* **1999**, *9*, 293–300.
57. Rasmussen, C.E.; Williams, C.K.I. *Gaussian Processes for Machine Learning*; The MIT Press: New York, NY, USA, 2006.
58. Lázaro-Gredilla, M.; Titsias, M.K.; Verrelst, J.; Camps-Valls, G. Retrieval of biophysical parameters with heteroscedastic Gaussian processes. *IEEE Geosci. Remote Sens. Lett.* **2014**, *11*, 838–842.
59. Rivera Caicedo, J.; Verrelst, J.; Muñoz-Marí, J.; Moreno, J.; Camps-Valls, G. Toward a semiautomatic machine learning retrieval of biophysical parameters. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2014**, *7*, 1249–1259.
60. Camps-Valls, G.; Gómez-Chova, L.; Muñoz-Marí, J.; Lázaro-Gredilla, M.; Verrelst, J. SimpleR: A Simple Educational Matlab Toolbox for Statistical Regression, V2.1. 2013. Available online: <http://www.uv.es/gcamps/software.html> (accessed on 6 September 2017).
61. Wold, H.O.A. Non-Linear Estimation by Iterative Least Squares Procedures. In *Research Papers in Statistics, Festschrift for J. Neyman*; David, F.J., Neyman, J., Eds.; Wiley: London, UK, 1966.
62. Van Der Tol, C.; Verhoef, W.; Timmermans, J.; Verhoef, A.; Su, Z. An integrated model of soil-canopy spectral radiances, photosynthesis, fluorescence, temperature and energy balance. *Biogeosciences* **2009**, *6*, 3109–3129.
63. Timmermans, J.; Su, Z.; Van Der Tol, C.; Verhoef, A.; Verhoef, W. Quantifying the uncertainty in estimates of surface-atmosphere fluxes through joint evaluation of the SEBS and SCOPE models. *Hydrol. Earth Syst. Sci.* **2013**, *17*, 1561–1573.
64. Duffour, C.; Oliosio, A.; Demarty, J.; Van der Tol, C.; Lagouarde, J.P. An evaluation of SCOPE: A tool to simulate the directional anisotropy of satellite-measured surface temperatures. *Remote Sens. Environ.* **2015**, *158*, 362–375.
65. Zhang, Y.; Guanter, L.; Berry, J.; Joiner, J.; van der Tol, C.; Huete, A.; Gitelson, A.; Voigt, M.; Kahler, P. Estimation of vegetation photosynthetic capacity from space-based measurements of chlorophyll fluorescence for terrestrial biosphere models. *Glob. Chang. Biol.* **2014**, *20*, 3727–3742.
66. Feret, J.B.; François, C.; Asner, G.P.; Gitelson, A.A.; Martin, R.E.; Bidel, L.P.R.; Ustin, S.L.; le Maire, G.; Jacquemoud, S. PROSPECT-4 and 5: Advances in the leaf optical properties model separating photosynthetic pigments. *Remote Sens. Environ.* **2008**, *112*, 3030–3043.
67. Rivera, J.; Sabater, N.; Tenjo, C.; Vicent, J.; Alonso, L.; Moreno, J. Synthetic scene simulator for hyperspectral spaceborne passive optical sensors. Application to ESA's FLEX/Sentinel-3 tandem mission. In Proceedings of the 6th Workshop on Hyperspectral Image and Signal Processing, Lausanne, Switzerland, 24–27 June 2014.
68. Conti, S.; O'Hagan, A. Bayesian emulation of complex multi-output and dynamic computer models. *J. Stat. Plan. Inference* **2010**, *140*, 640–651.
69. Verrelst, J.; Rivera, J.; Moreno, J.; Camps-Valls, G. Gaussian processes uncertainty estimates in experimental Sentinel-2 LAI and leaf chlorophyll content retrieval. *ISPRS J. Photogram. Remote Sens.* **2013**, *86*, 157–167.
70. Lecun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444.
71. Schmidhuber, J. Deep Learning in neural networks: An overview. *Neural Netw.* **2015**, *61*, 85–117.
72. Vicent, J.; Sabater, N.; Verrelst, J.; Alonso, L.; Moreno, J. Assessment of approximations in aerosol optical properties and vertical distribution into FLEX atmospherically-corrected surface reflectance and retrieved Sun-Induced Fluorescence. *Remote Sens.* **2017**, *9*, 675.
73. Mousivand, A.; Menenti, M.; Gorte, B.; Verhoef, W. Multi-temporal, multi-sensor retrieval of terrestrial vegetation properties from spectral-directional radiometric data. *Remote Sens. Environ.* **2015**, *158*, 311–330.
74. Saltelli, A.; Ratto, M.; Andres, T.; Campolongo, F.; Cariboni, J.; Gatelli, D.; Saisana, M.; Tarantola, S. *Global Sensitivity Analysis: The Primer*; John Wiley & Sons: Chichester, UK, 2008.
75. Nossent, J.; Elsen, P.; Bauwens, W. Sobol'sensitivity analysis of a complex environmental model. *Environ. Model. Softw.* **2011**, *26*, 1515–1525.

76. Jacquemoud, S.; Baret, F.; Andrieu, B.; Danson, F.; Jaggard, K. Extraction of vegetation biophysical parameters by inversion of the PROSPECT + SAIL models on sugar beet canopy reflectance data. Application to TM and AVIRIS sensors. *Remote Sens. Environ.* **1995**, *52*, 163–172.
77. Kuusk, A. Monitoring of vegetation parameters on large areas by the inversion of a canopy reflectance model. *Int. J. Remote Sens.* **1998**, *19*, 2893–2905.
78. Myneni, R.; Hoffman, S.; Knyazikhin, Y.; Privette, J.; Glassy, J.; Tian, Y.; Wang, Y.; Song, X.; Zhang, Y.; Smith, G.; et al. Global products of vegetation leaf area and fraction absorbed PAR from year one of MODIS data. *Remote Sens. Environ.* **2002**, *83*, 214–231.
79. Richter, R. A spatially adaptive fast atmospheric correction algorithm. *Int. J. Remote Sens.* **1996**, *17*, 1201–1214.
80. Thome, K.; Palluconi, F.; Takashima, T.; Masuda, K. Atmospheric correction of ASTER. *IEEE Trans. Geosci. Remote Sens.* **1998**, *36*, 1199–1211.
81. Kotchenova, S.; Vermote, E.; Matarrese, R.; Klemm, F., Jr. Validation of a vector version of the 6S radiative transfer code for atmospheric correction of satellite data. Part I: Path radiance. *Appl. Opt.* **2006**, *45*, 6762–6774.
82. Guanter, L.; Alonso, L.; Moreno, J. CHRIS Proba Atmospheric Correction Module. In *Algorithm Theoretical Basis Document*; ESA ESRIN Contract No. 20442; ESA ESRIN: Frascati, Italy, 2007.
83. Hagolle, O.; Huc, M.; Pascual, D.; Dedieu, G. A multi-temporal and multi-spectral method to estimate aerosol optical thickness over land, for the atmospheric correction of FormoSat-2, LandSat, VEN $\mu$ S and Sentinel-2 images. *Remote Sens.* **2015**, *7*, 2668–2691.
84. Lefèvre, M.; Oumbe, A.; Blanc, P.; Espinar, B.; Gschwind, B.; Qu, Z.; Wald, L.; Schroedter-Homscheidt, M.; Hoyer-Klick, C.; Arola, A.; et al. McClear: A new model estimating downwelling solar radiation at ground level in clear-sky conditions. *Atmos. Meas. Tech.* **2013**, *6*, 2403–2418.
85. Richter, R.; Heege, T.; Kiselev, V.; Schläpfer, D. Correction of ozone influence on TOA radiance. *Int. J. Remote Sens.* **2014**, *35*, 8044–8056.



© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).