

# PREDICTING YEAR OF PLANTATION WITH HYPERSPECTRAL AND LIDAR DATA

*Adrià Descals, Luis Alonso and Gustau Camps-Valls*

<sup>1</sup>Image Processing Laboratory (IPL), Universitat de València, Spain

## ABSTRACT

This paper introduces a methodology for predicting the year of plantation (YOP) from remote sensing data. The application has important implications in forestry management and inventorying. We exploit hyperspectral and LiDAR data in combination with state-of-the-art machine learning classifiers. In particular, we present a complete processing chain to extract spectral, textural and morphological features from both sensory data. Features are then combined and fed a Gaussian Process Classifier (GPC) trained to predict YOP in a forest area in North Carolina (US). The GPC algorithm provides accurate YOP estimates, reports spatially explicit maps and associated confidence maps, and provides sensible feature rankings.

**Index Terms**— Vegetation monitoring, Hyperspectral, LiDAR, Data fusion, Gaussian process classification

## 1. INTRODUCTION

*‘How old would you be if you didn’t know how old you are?’*  
— Satchel Paige

The fundamentals of forest inventories based on field sampling were developed in the early twentieth century. These conventional inventories rely heavily on field measurements, and its practice has been in place even in recent national inventories. In contrast to field-based inventories, airborne laser scanning (ALS) supported inventories attained outstanding results [1, 2], mainly due to the high precision of LiDAR sensor and its cost effectiveness. Consequently, in the last decade, the interest in LiDAR for forest applications has grown a lot and has become a field of active research.

Several studies have determined a correlation between dendrometric variables such as the normal diameter or crown biomass, and the height measurements taken from Aerial Laser Scanning (ALS) data [2, 3]. This allows not only an estimation of stand variables at large-scale forest but also a proxy for the timber stock and wood quality features. Recently, other studies include the fusion of LiDAR and HSI for forestry applications. HSI has been commonly used (alone)

for health monitoring and biophysical parameter retrieval [4]. Certainly, the fusion of optical (hyperspectral) data with LiDAR can be beneficial for either retrieval of biophysical parameters and for complex classification problems, as successfully reported in previous works [5, 6]. Forest inventories provide data and information at tree and stand level that support the forest management goals. Stand age, among other variables, is measured in the inventory to assess the current status of the forest. Although stand age is often considered irrelevant in productive forests, the year of plantation (YOP) might give a quick overview about the present situation of the forest. Stand age can be also considered a handy information that suggests when forest practices are required, such as thinning or pruning. In addition, since stand age is a driver of tree height, this relationship is used to quantify the site index and determine the rotation cycle.

In this paper, we aim to exploit the capabilities of machine learning classifiers to determine the year of plantation using both LiDAR and hyperspectral data. In particular we will focus on Gaussian Process Classification (GPC) [7]. Gaussian processes are rooted in Bayesian inference and typically provide state-of-the-art accuracy results, accompanied with several interesting properties: 1) unlike other classifiers such as support vector machines (SVMs) [8, 9], 2) they can provide confidence intervals for the predictions, 3) hyperparameters can be learned by approximate inference, and 4) they yield a feature ranking by looking at the variable-dependent hyperparameters. We have recently reported all these properties in the context of parameter retrieval in [10], but no attention has been paid to them in the context of classification problems so far, in which we will focus in this paper. Note that our temporal resolution for YOP prediction is yearly so classification is a preferred (and more convenient) setup for the problem.

The remainder of the paper is organized as follows. Section §2 details both the data collection and the proposed processing chain. Section §3 describes Gaussian Processes for classification (GPC), its adaptation to multiclass settings, and the proposed technique to derive feature rankings from the trained models. Experimental evidence of performance of our proposal is given in §4. We conclude in §5 with some remarks and outline future work.

## 2. DATA COLLECTION AND PROCESSING CHAIN

This section reviews the data collected and the processing chain proposed for LiDAR and hyperspectral feature extraction and data fusion.

---

The research was funded by the European Research Council (ERC) under the ERC-CoG-2014 SEDAL project (grant agreement 647423), the Spanish Ministry of Economy and Competitiveness (MINECO) through the project TIN2015-64210-R, and the FLEX-US Project, an ESA/NASA Joint Campaign for the Deployment of the Airborne HyPlant Imaging Spectrometer (ESA Contract No.4000109199/13/NL/FF/ff). We would like to thank the Goddard group that provided the data: NASA Goddard Space Flight Center (GSFC), Greenbelt, MD 20771, USA.

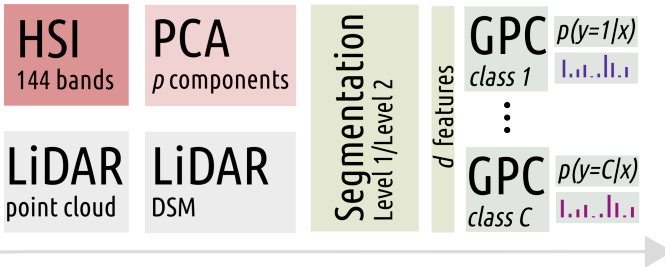
## 2.1. Data collection

The data used in this work were collected by the airborne Goddard’s LiDAR, Hyperspectral, and Thermal imager<sup>1</sup> (G-LiHT) [11], during the ESA/NASA Joint FLEX-US (FLuorescence EXplorer experiment in USA) campaign from October 24th to 27th, 2013. G-LiHT consists of three sensors: (1) a small footprint profiling and scanning LiDAR, (2) a Hyperspec VNIR Concentric Imaging Spectrometer that records hyperspectral images with 114 bands distributed in the VNIR and (3) a Gobi-384 sensor for thermal imagery. This study uses the LiDAR cloud point and the hyperspectral image generated in the VNIR.

The study area covers 490 ha of forest in Parker Track, North Carolina (USA) (35° 49’ N, 76° 39’ W). The forest is divided in nine logging units which are managed as even-aged stands of loblolly pine (*Pinus taeda*) for timber production. Therefore, trees within the same unit have similar *dendrological* features, such as height and normal diameter. The year of plantation for this forest stands is 1984, 1986, 1990, 1992, 2001, 2009, 2010, 2011, and 2013. In addition, some areas were left as conservation sites. In these areas, the forest composition is a mixture of loblolly pine and deciduous trees, and tree measures are heterogeneous.

## 2.2. Data processing chain

The proposed processing chain in this work is shown in Fig. 1. The first step involves a dimensionality reduction of the hyperspectral data through PCA, and the calculation of the Digital Surface Model (DSM) from LiDAR point cloud data. Afterwards, a multi-level segmentation is performed on top of both pieces of information. The second part of the chain focuses on ranking these features from the embedded hyperparameters in the GPC model.



**Fig. 1:** Proposed processing chain: HSI is reduced to a number of  $p$  components using PCA, while we extract a DSM from the LiDAR data. Then a multi-level segmentation is performed with morphological filters. These feature extraction yields a set of  $d$  features used in a one-versus-all GPC classifier scheme, that not only provide a full prediction probability but also feature rankings per class.

Segmentation is the process of grouping pixels into multiple segments in such a way that the new values are more meaningful and easier to analyze. We used a bottom-up merging algorithm [12], based on spatial and spectral homogeneity criteria, to produce a two-level segmentation, level 1 (L1)

<sup>1</sup><http://gliht.gsfc.nasa.gov/>

and level 2 (L2). The segmentation was carried out using a 2-meter DSM derived from the LiDAR cloud point. We set the segmentation parameters in L1 to achieve a delimitation of single tree crowns, while L2 was produced with a coarser segmentation in order to clump trees with similar heights.

The two segmentation levels served to extract features from the hyperspectral image and the LiDAR point cloud. The hyperspectral image was transformed using a Principal Component Analysis (PCA) for dimensionality reduction, projecting the hyperspectral image onto  $p = 10$  components. For the first ten components, the mean and standard deviation were extracted considering the pixels included in one image object. Similarly, we derived height percentiles and intensity values considering those LiDAR returns whose ground projection was contained within the same object. Finally, we extracted texture features by computing gray-level cooccurrence matrix (GLCM) statistics using the DSM values enclosed in an object.

## 3. GAUSSIAN PROCESS CLASSIFICATION (GPC)

Gaussian processes (GPs) are Bayesian state-of-the-art tools for discriminative machine learning. GPs were first proposed in statistics [13], and they are well-known to the geostatistics community as kriging. GP algorithms have been proposed for regression, classification and dimensionality reduction [7].

### 3.1. Classification with Gaussian Processes

Notationally, we are given a set of input-output data pairs  $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$ , where  $\mathbf{x} := [x_1, \dots, x_d] \in \mathbb{R}^d$  and  $y \in \{0, 1\}$ . Standard regression approximates the observations (often referred to as outputs, targets or labels)  $\{y_i\}_{i=1}^N$  as the sum of some unknown latent function  $f(\mathbf{x})$  of the inputs  $\{\mathbf{x}_i \in \mathbb{R}^d\}_{i=1}^N$  plus Gaussian noise, i.e.  $y_i = f(\mathbf{x}_i) + e_i$ , where  $e_i \sim \mathcal{N}(0, \sigma_n^2)$ . GP models proceeds in a Bayesian, non-parametric way, to fit the observed data. For the case of GP classification, we need to proceed by placing a GP prior over the latent function  $f(\mathbf{x})$  and then transform it with a logistic function to obtain a prior on  $p(y = 1|f(\mathbf{x})) = \sigma(f(\mathbf{x})) = (1 + \exp(-f(\mathbf{x})))^{-1}$ . This response function  $\sigma(\cdot)$  “squashes” the real-valued latent function  $f$  into a  $(0, 1)$ -interval representing the posterior probability for  $y$  [7].

Let us now define  $\mathbf{f} = [f(\mathbf{x}_1), \dots, f(\mathbf{x}_N)]^\top$ , the matrix of training input data  $\mathbf{X} \in \mathbb{R}^{N \times d}$ , all training output labels in  $\mathbf{y} = [y_1, \dots, y_N]^\top$ , and let us denote the training dataset as  $\mathcal{D} \sim \mathbf{X}, \mathbf{y}$ . Inference is naturally divided into two steps: first computing the distribution of the latent variable corresponding to a test example  $\mathbf{x}_*$

$$p(f_*|\mathcal{D}, \mathbf{x}_*) = \int p(f_*|\mathbf{X}, \mathbf{x}_*, \mathbf{f})p(\mathbf{f}|\mathcal{D})d\mathbf{f},$$

where  $p(\mathbf{f}|\mathcal{D}) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{X})/p(\mathbf{y}|\mathbf{X})$  is the posterior over the latent variables, and subsequently using this distribution over the latent  $\mathbf{f}$  to produce a probabilistic prediction

$$p(y_* = 1|\mathcal{D}, \mathbf{x}_*) = \int \sigma(f_*)p(f_*|\mathcal{D}, \mathbf{x}_*)df_*.$$

This integral is however not analytically tractable so one has to resort numerical methods or approximations to solve it [7]. One could simply resort to Markov Chain Monte Carlo (MCMC) methods, but they are computationally too expensive. Alternatively, by assuming a Gaussian approximation to the posterior given the hyperparameters  $\theta$ ,  $p(\mathbf{f}|\mathbf{X}, \mathbf{y}, \theta) \approx q(\mathbf{f}|\mathbf{X}, \mathbf{y}, \theta)\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ , one can use two convenient alternatives to estimate the predictive mean  $\boldsymbol{\mu}$  and variance  $\boldsymbol{\Sigma}$ : the Laplace approximation (LA) or the expectation propagation (EP) [14] method. We used the EP approximation that has revealed a more accurate approximation [15].

### 3.2. Inference with the GPC

Note that the GPC yields a full posterior predictive distribution over  $y_*$ : this allows us to obtain not only mean predictions for test data,  $\mu_{f_*}$ , but also the so-called “error-bars”,  $\sigma_{f_*}^2$ , assessing the uncertainty of the mean prediction. These are the so-called approximate *predictive mean* and *predictive variance* for the latent variable  $\mathbf{f}$ . The EP approximation gives a Gaussian approximation to the posterior distribution,  $p(f_*|\mathcal{D}, \mathbf{x}_*) = \mathcal{N}(y_*|\mu_{f_*}, \sigma_{f_*}^2)$ , with predictive mean and variance:

$$\begin{aligned}\mu_{f_*} &= \mathbf{k}_{f_*}^\top (\tilde{\mathbf{K}}_{\text{ff}} + \tilde{\boldsymbol{\Sigma}})^{-1} \mathbf{y} = \mathbf{k}_{f_*}^\top \boldsymbol{\alpha} \\ \sigma_{f_*}^2 &= \sigma^2 + k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_{f_*}^\top (\tilde{\mathbf{K}}_{\text{ff}} + \tilde{\boldsymbol{\Sigma}})^{-1} \mathbf{k}_{f_*},\end{aligned}$$

where  $\mathbf{k}_{f_*} = [k(\mathbf{x}_*, \mathbf{x}_1), \dots, k(\mathbf{x}_*, \mathbf{x}_n)]^\top$  indicates the vector of covariances between the test point  $\mathbf{x}_*$  and the  $N$  training points, and  $\boldsymbol{\alpha}$  is the solution weight vector. Approximating the predictive distribution for the binary targets,  $q(y_* = 1|\mathcal{D}, \mathbf{x}_*)$ , requires averaging the output of all possible models w.r.t. the Gaussian posterior before, which can be done analytically [7][sect. 3.6.1]. We followed a one-against-all (OAA) classification scheme and the maximum vote was weighted according to the class-wise confidence level. While we are aware of more sophisticated multiclass schemes, the OAA scheme allowed us to derive intuitive confidence intervals and identify the most difficult YOPs in the dataset easily.

### 3.3. Ranking features with GPC

We selected the anisotropic squared exponential (SE) kernel function  $K$  given by:

$$K(\mathbf{x}, \mathbf{x}') = \nu \exp\left(-\sum_{k=1}^d \frac{(x_k - x'_k)^2}{2\sigma_k^2}\right), \quad (1)$$

where  $\nu$  is a scaling factor, and we have one  $\sigma_k$  per each input features,  $k = 1, \dots, d$ . This is a very flexible covariance function that typically suffices to tackle most of the problems. Furthermore, this kernel function is also used for the so-called *automatic relevance determination* (ARD) [16], i.e., studying the influence of each input component to the output. We will use this to gain some knowledge about the relative relevance of input features (LiDAR and hyperspectral) included in  $\mathbf{x}$ .

## 4. EXPERIMENTAL RESULTS

We show here the obtained results in the YOP estimation from remote sensing data. We first describe the model development, and then analyze the results both in terms of accuracy and interpretability.

### 4.1. Model development and results

The training and testing values were obtained by selecting objects in the segmentation L2. A total of 200 segments were chosen for training and 1500 for testing. The classification models were applied considering three experiments: (1) only LiDAR features, (2) only HSI features, and (3) combining LiDAR and HSI features. A numerical comparison is shown in Table 1. It becomes clear that the best results are obtained when combining both pieces of information, and that LiDAR features are more discriminative than HSI features. Figure 2 shows a simple analysis of the results over an image chip: the classification map and associated confidence map for the different classes give evidence of accurate YOP prediction.

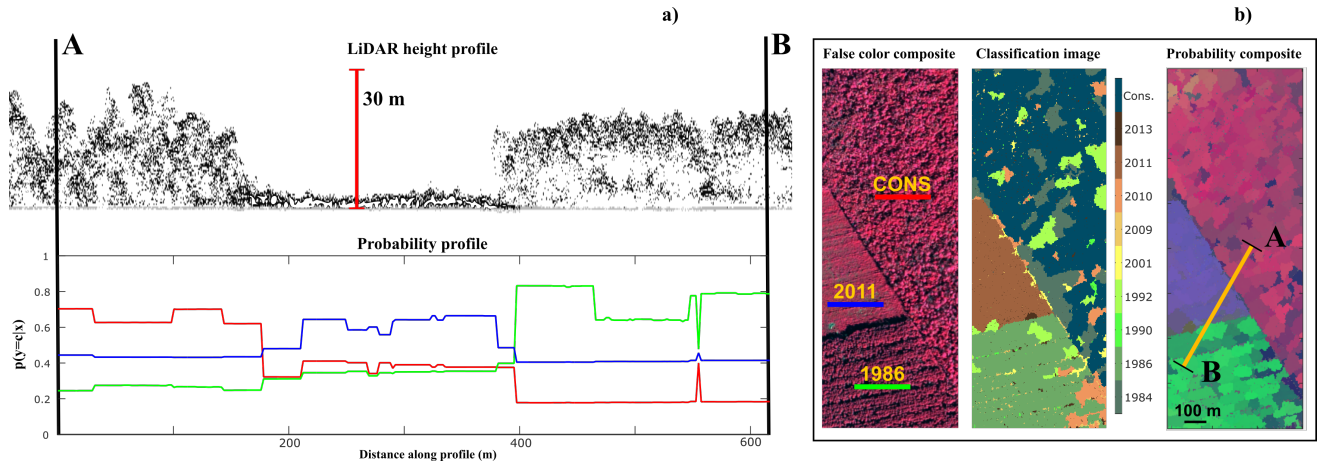
	HSI	LiDAR	LiDAR+HSI
Cohen’s $\kappa$ statistic	0.43	0.54	0.61
Overall accuracy, OA[%]	49.48	59.22	65.16

**Table 1:** Results of the classification with different input features.

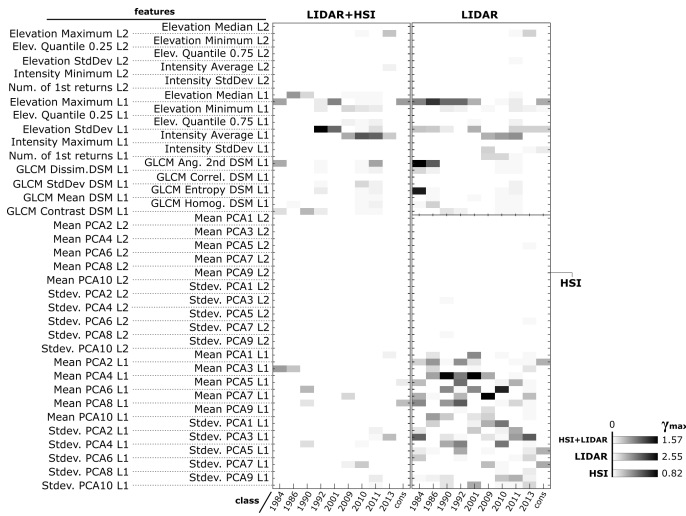
### 4.2. Feature ranking

The gamma values  $\gamma_k \propto 1/\sigma_k^2$  obtained from the GPC models are given in Fig. 3. High  $\gamma_k$  values indicate high relevance of the particular feature in the classification model. When only LiDAR features were considered, ‘Elevation maximum in L2’ and two texture features help distinguishing mature forest stands against the rest of YOP classes. In younger ages, the height is not so determining to discriminate among early plantations, so height features lose relevance. Most of the features obtained from the segmentation level L1 appear to be irrelevant. This suggests that a single-tree crown segmentation might be unnecessary, in contrast to a coarser segmentation – in L2 – in which features characterize clumps of trees that are representative of the forest age.

When the model requires a higher amount of variables for distinguishing a class, we can presume that the model was not able to detect the class satisfactorily. In such case, the gamma values are low and close between them. This is generally the case for young stands, and for most of the classes in the experiment where only HSI features were considered. The low gammas indicate that HSI features do not perform well solely, while LiDAR features may be considered alone particularly for classifying mature forest. In the experiment LiDAR+HSI, LiDAR features seem to be most relevant, but HSI features complement in certain classes. For example, a mature forest such as class *1986* requires two features only: ‘Elevation median’ and ‘Mean PCA3’. Other classes might need more features (e.g. younger ages), but the relevant features remain among HSI and LiDAR.



**Fig. 2:** Spatial analysis of the results: (a) the LiDAR height profile for a given transect A-B (top) and the predictive variance (posterior probability) of the GPC for the classes *conservation*, 1986, and 2011. (b) for a specific area; RGB false color composite, along with the GPC classification map, and RGB composite using the probability (confidence interval) images of *conservation* (R), 1986 (G), and 2011 (B).



**Fig. 3:** Gamma values for the 68 features. The experiments consider 1. LiDAR, 2. HSI, and 3. LiDAR + HSI features. The legend depicts the range for gamma in each experiment.

## 5. CONCLUSIONS

This paper introduced a methodology for predicting the YOP from hyperspectral and LiDAR data. After an exhaustive feature extraction (involving spectral, textural and morphological features) we used advanced GPC for classification. We illustrated the results in a forest area in North Carolina (US). The GPC algorithm showed improved results when combining hyperspectral and LiDAR data. Beyond accuracy, we paid special attention to the feature ranking after analyzing the associated lengthscales in the kernel. Height and texture features appeared to be relevant features specially for predicting the YOP in mature forests. Future work will consider extending the formulation of GPC to cope with spatial-spectral relations directly in the kernel.

## 6. REFERENCES

- [1] L. M. Moskal, *et al.*, *Proceedings of Silvilaser* pp. 154–163 (2009).
- [2] M. Maltamo, E. Næsset, J. Vauhkonen, *Concepts and case studies. Manag For Ecosys* **27**, 2014 (2014).
- [3] S. C. Popescu, R. H. Wynne, R. F. Nelson, *Canadian journal of remote sensing* **29**, 564 (2003).
- [4] C.-I. Chang, *Hyperspectral data exploitation: theory and applications* (John Wiley & Sons, 2007).
- [5] M. Dalponte, L. Bruzzone, D. Gianelle, *IEEE Transactions on Geoscience and Remote Sensing* **46**, 1416 (2008).
- [6] M. Campos-Taberner, *et al.*, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* (2016).
- [7] C. E. Rasmussen, C. K. I. Williams, *Gaussian Processes for Machine Learning* (The MIT Press, New York, 2006).
- [8] G. Camps-Valls, L. Bruzzone, *Kernel Methods for Remote Sensing Data Analysis* (John Wiley and Sons, 2009).
- [9] G. Camps-Valls, D. Tuia, L. Gmez-Chova, S. Jimnez, J. Malo, eds., *Remote Sensing Image Processing* (Morgan & Claypool Publishers, LaPorte, CO, USA, 2011).
- [10] G. Camps-Valls, *et al.*, *IEEE Geoscience and Remote Sensing Magazine* (2016).
- [11] B. D. Cook, *et al.*, *Remote Sensing* **5**, 4045 (2013).
- [12] U. C. Benz, P. Hofmann, G. Willhauck, I. Lingenfelder, M. Heynen, *ISPRS Journal of photogrammetry and remote sensing* **58**, 239 (2004).
- [13] A. O’Hagan, J. F. C. Kingman, *Journal of the Royal Statistical Society. Series B (Methodological)* **40**, 1 (1978).
- [14] T. Minka, *Proceedings of the Seventeenth Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-01)* (Morgan Kaufmann, San Francisco, CA, 2001), pp. 362–369.
- [15] M. Kuss, C. E. Rasmussen, *J. Mach. Learn. Res.* **6**, 1679 (2005).
- [16] C. Bishop, *Pattern Recognition and Machine Learning* (Springer, 2006).