Supplementary material

Mid-infrared Laser Spectroscopy Detection and Quantification of Explosives in Soils Using Multivariate Analysis and Artificial Intelligence

Leonardo C. Pacheco-Londoño [1,2,*], Eric Warren [1], Nataly J. Galán-Freyle [1,2], Reynaldo Villarreal-González [2], Joaquín A. Aparicio-Bolaño [3,4], María L. Ospina-Castro [5], Wei-Chuan Shih [6] and Samuel P. Hernández-Rivera [1,*]

[1] ALERT DHS Center of Excellence for Explosives Research, Department of Chemistry, University of Puerto Rico, Mayagüez, PR 00681, USA; leonardo.pacheco@upr.edu (L.P.); nataly.galan@upr.edu (N.G.); eric-warren@uiowa.edu (E.W.)

[2] MacondoLab, Universidad Simón Bolívar, Barranquilla, Colombia leonardo.pacheco@unisimonbolivar.edu.co (L.P.); nataly.galan@unisimonbolivar.edu.co (N.G.); rvillarreal2@unisimonbolivar.edu.co (R.V.)

[3] Department of Physics, Florida International, FL, USA; aparicio_joaquin@hotmail.com

[4] Physics, Chemistry, Physics and Earth Sciences Department, Miami-Dade College, Kendall Campus, Miami, FL, USA; aparicio_joaquin@hotmail.com

[5] Grupo de Investigación Química Supramolecular Aplicada. Programa de Química, Universidad del Atlántico, Barranquilla, Colombia; mariaospina@mail.uniatlantico.edu.co

[6] Department of Electrical & Computer Engineering University of Houston, 4800 Calhoun Rd. Eng. Bldg. 1, Rm. N308, Houston TX 77204-4005, USA; wshih@central.uh.edu

* Correspondence: samuel.hernandez3@upr.edu; Tel.: 787-265-5404; 787-832-4040, x-5458, X-2638 (S.H.) leonardo.pacheco@unisimonbolivar.edu.co; Tel.: +57-304-648-9549 (L.P.)

Table S1. Composition percent of soil (NAT-S) samples.

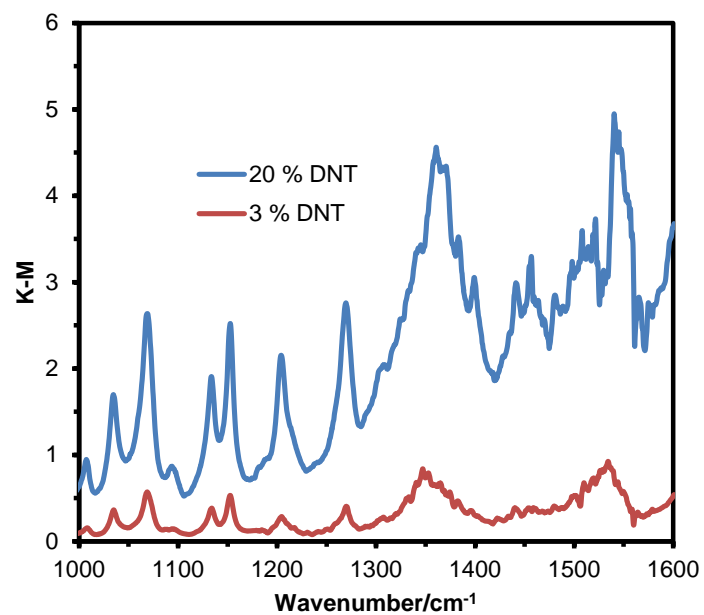|	TGA	H$_2$O$_2$ Method	TDS	Total Analyses
water content	5.3%			5.3%
OM		4.1%		4.1%
non-volatile solids	85.6%			85.6%
volatile				5.0%
volatile + OM	9.1%			9.1%
SSW			13.1%	13.1%
non-soluble solids				77.4%
Total solids				90.6%

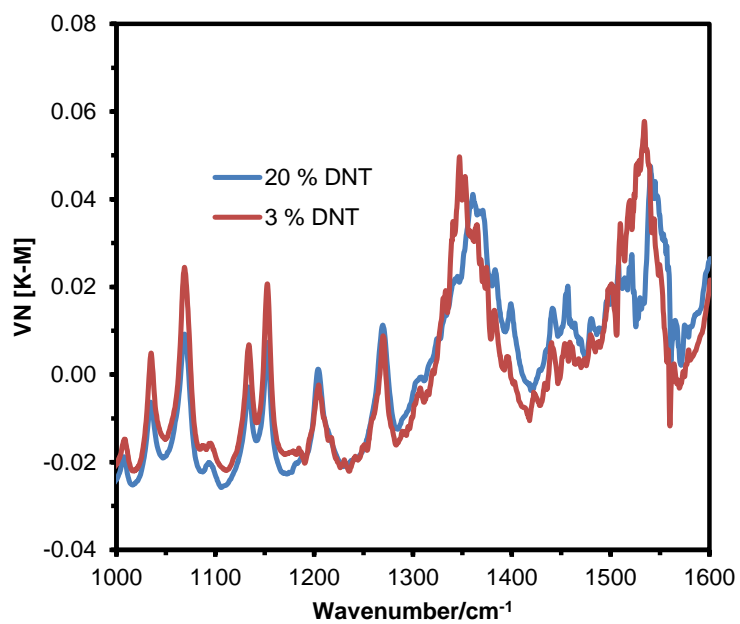**Figure S1.** KM Spectra of 20 and 3 % of DNT from the NAT-S sample.

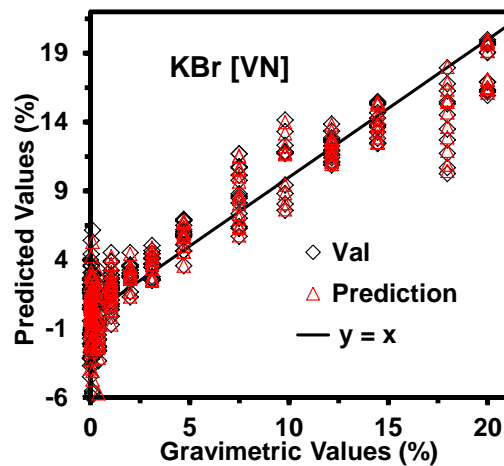**Figure S2.** Vector Normalization of KM Spectra of 20 and 3 % of DNT from NAT-S sample.

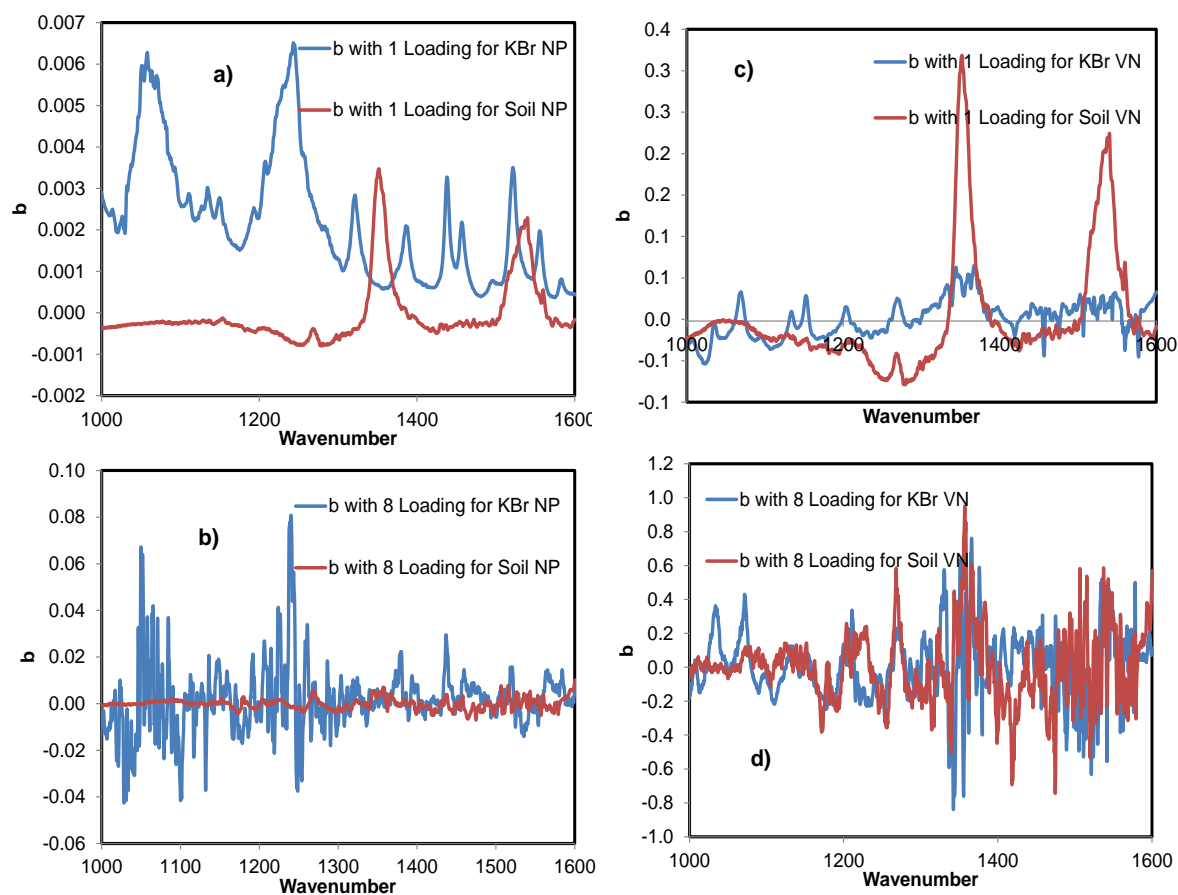**Figure S3.** PLS models for DNT in KBr using VN prepossessing.

Figure S4. a) **b** regression vector for models without preprocessing and with one loading, b) **b** regression vector for models with vector normalization preprocessing and with one loading, c) **b** regression vector for models without preprocessing and with eight loadings, d) **b** regression vector for models with vector normalization preprocessing and with eight loading
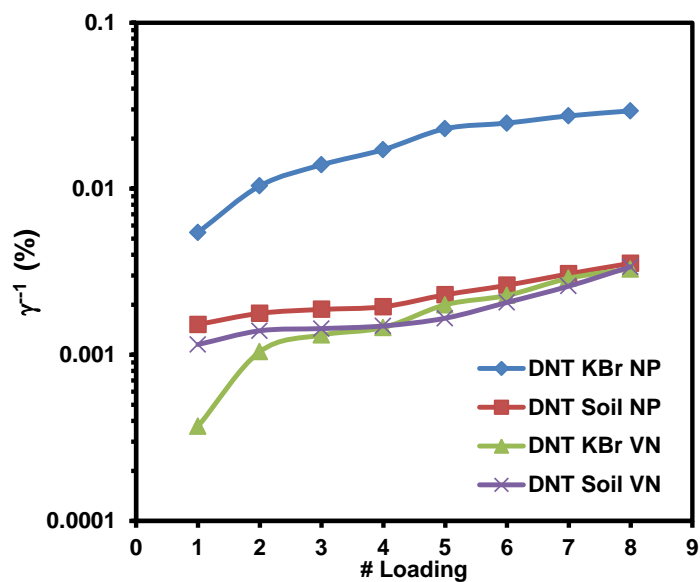
Figure S5. Plot of # of loading vs. analytical sensitivity for PLS models for DNT in KBr and Soil with VN preprocessing and without preprocessing.
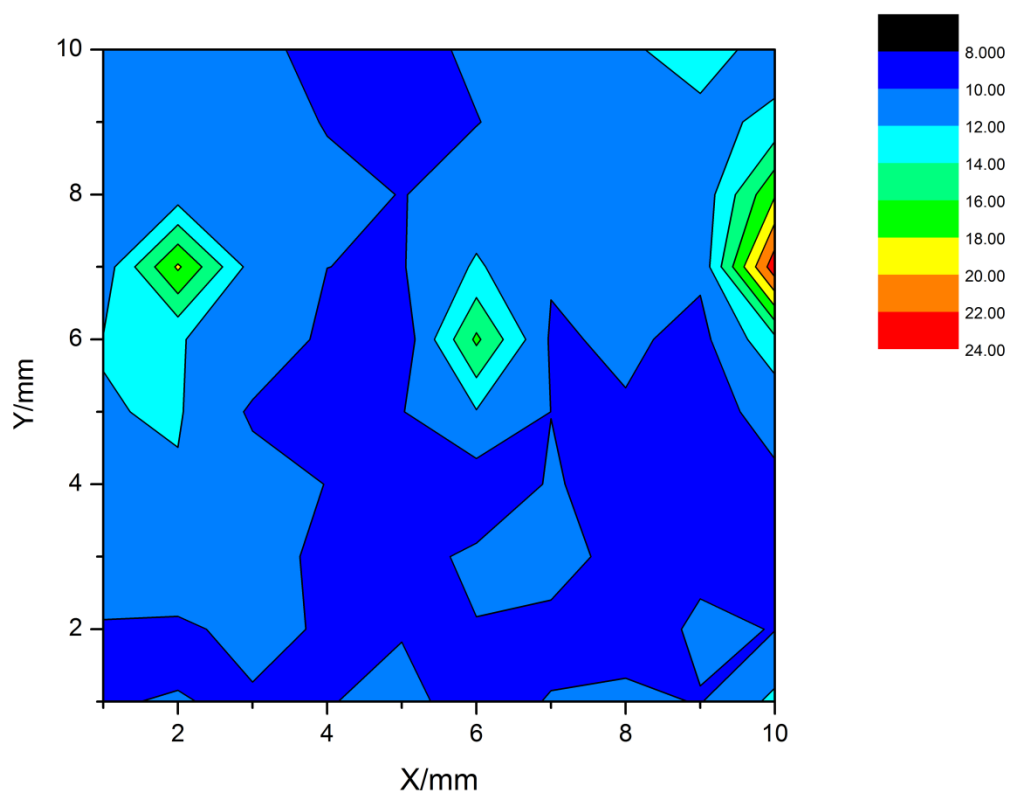

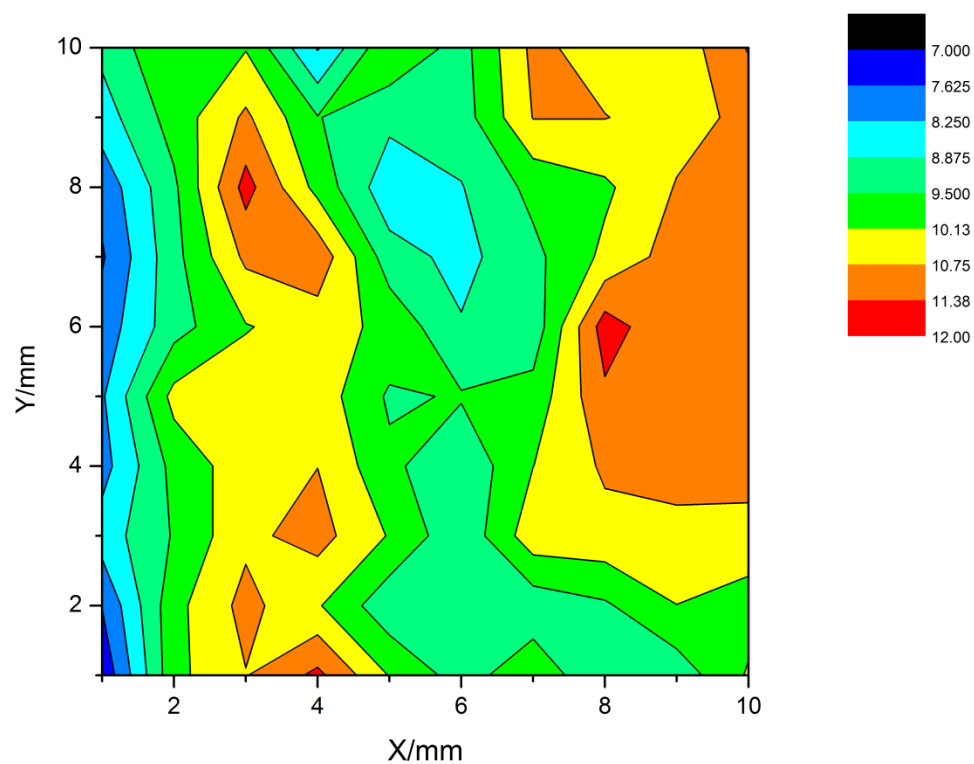**Figure S6**. Map of % of DNT from NAT-S model (VN) of NAT-S-M sample.

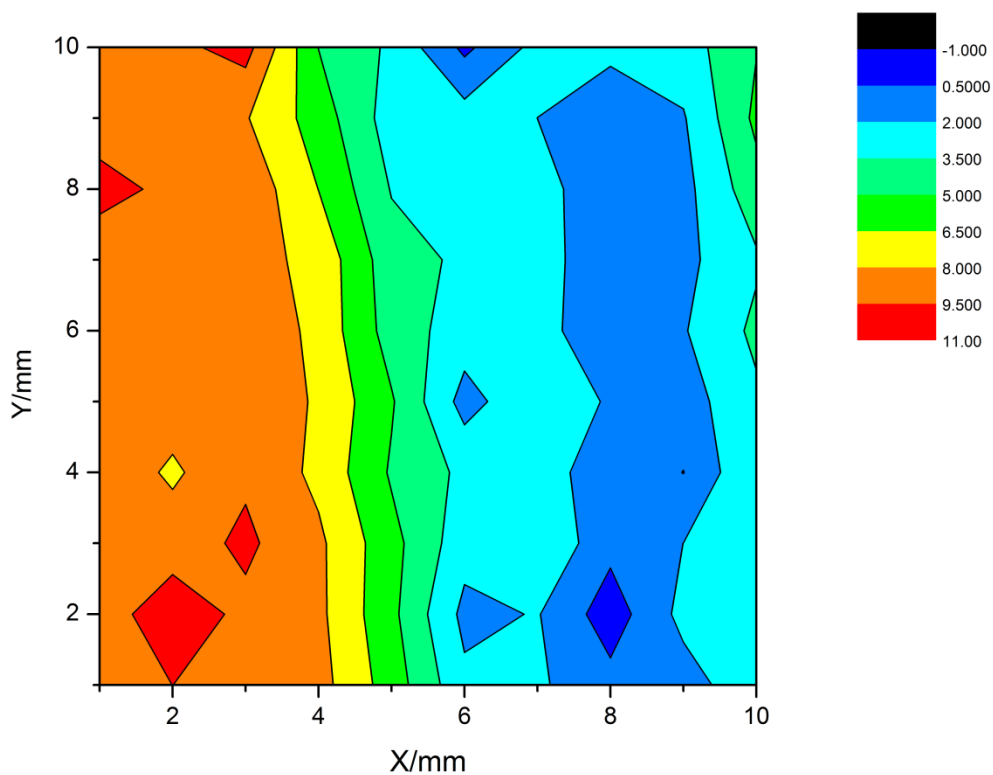**Figure S7.** Map of % of DNT from NAT-S model (VN) of NAT-S-MM sample

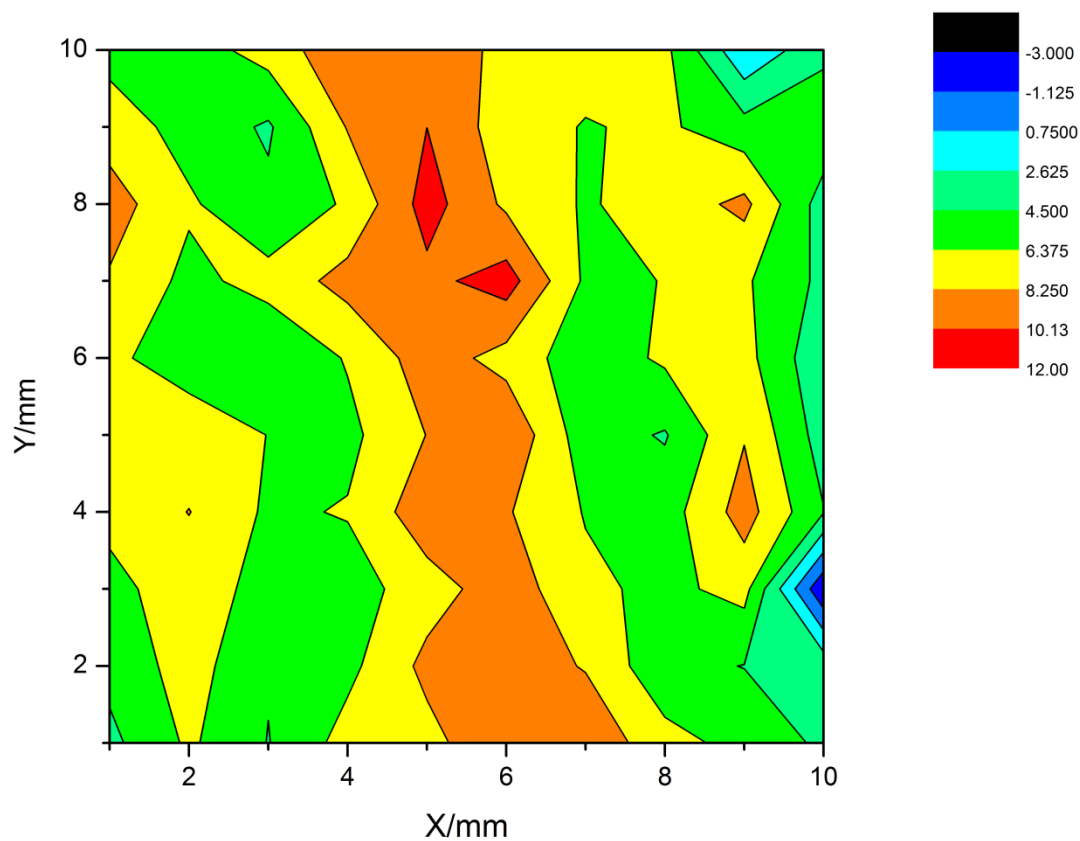**Figure S8.** Map of % of DNT from NAT-S model (VN) of NAT2-S sample.

Figure S9. Map of % of DNT from NAT-S model (VN) of NAT2-S sample
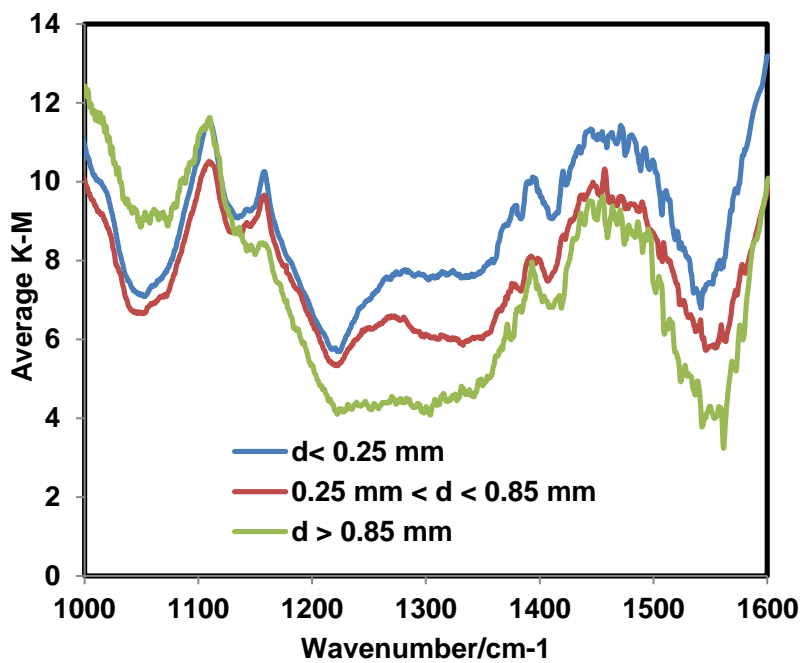

Figure S10. QCL Average spectra for different size of soil
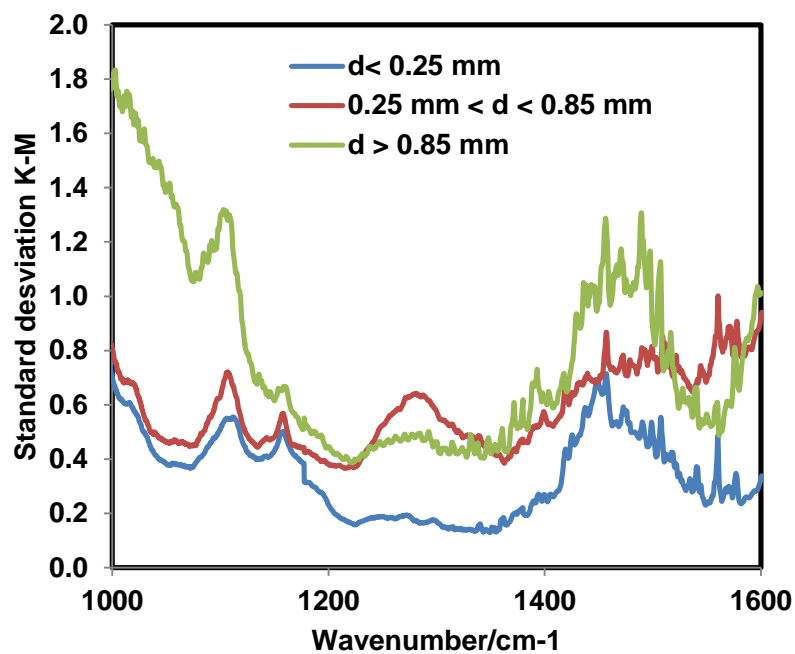
Figure S11. QCL spectra showing a standard deviation for various sizes of soil samples.
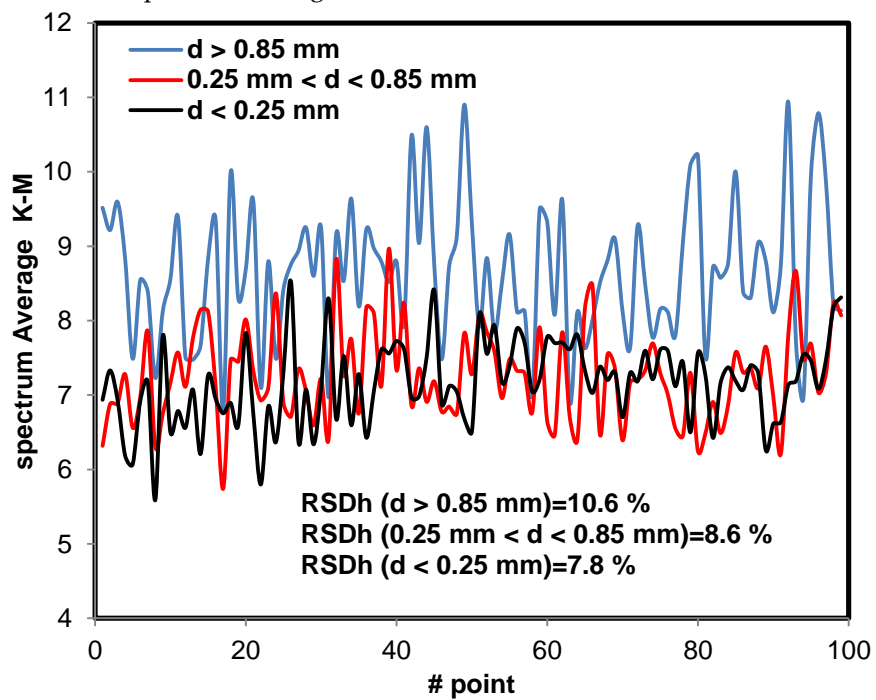

RSDh (d > 0.85 mm)=10.6 %
RSDh (0.25 mm < d < 0.85 mm)=8.6 %
RSDh (d < 0.25 mm)=7.8 %

Figure S12. QCL average spectrum for various sizes of soil.

Table S13. Basic description of matching learning methods for classification [60]

ML METHODS	DESCRIPTION
K-neighbors classifier(KNC)	Scikit-learn implements a K-neighbors classifier, a neighbors-based classification, where k is an integer value specified by the user. This is a type of instance-based learning or non-generalizing learning: it does not attempt to construct a general internal model, but simply stores instances of the training data. Classification is computed from a simple majority vote of the nearest neighbors of each point: a query point is assigned the data class, which has the most representatives within the nearest neighbors of the point.
SVC	Support vector machines for classification (SVC) is an algorithm capable of performing multi-class classification on a dataset. They are a set of supervised learning methods used for classification. These are based on the library (libsvm). In SVC, the fit time scales at least quadratically with the number of samples and may be impractical beyond tens of thousands of samples.
Decision Tree Classifier (DTC)	Decision Tree Classifier is a non-parametric supervised learning method is an algorithm capable of performing multi-class classification on a dataset. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. For instance, classical decision trees learn from data to approximate a sine curve with a set of if-then-else decision rules. The deeper the tree, the more complex the decision rules, and the fitter the model.
Random Forest Classifier (RFC)	Random forests classifier is an ensemble learning method for classification that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes of the individual trees. Random decision forests correct for decision trees' habit of overfitting to their training set. A random forest is a meta estimator that fits several decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.
AdaBoost Classifier (ABC)	An AdaBoost[61] classifier is a meta-estimator that begins by fitting a classifier on the original dataset and then fits additional copies of the classifier on the same dataset but where the weights of incorrectly classified instances are adjusted such that subsequent classifiers focus more on severe cases. This class implements the algorithm known as AdaBoost-SAMME[61].
Gaussian Naive Bayes (GNB)	Gaussian Naive Bayes (GaussianNB), implements the Gaussian Naive Bayes algorithm for classification. The likelihood of the features is assumed to be Gaussian. Can perform online updates to model parameters via partial_fit. For details on the algorithm used to update

	feature means and variance online, see Stanford CS tech report STAN-CS-79-773 by Chan, Golub, and LeVeque
Linear Discriminant Analysis (LDA)	Linear Discriminant Analysis, it is a classifier with a linear decision boundary, generated by fitting class conditional densities to the data and using Bayes' rule. The model fits a Gaussian density to each class, assuming that all classes share the same covariance matrix. The fitted model can also be used to reduce the dimensionality of the input by projecting it to the most discriminative directions.
Quadratic Discriminant Analysis (QDA)	Quadratic Discriminant Analysis, it is a classifier with a quadratic decision boundary, generated by fitting class conditional densities to the data and using Bayes' rule. The model fits a Gaussian density to each class.