



Earth Observation Image Librarian
IMAGE INFORMATION MINING
ALGORITHM DESIGN

Prepared By

Shiyong Cui

Corneliu Octavian Dumitru

Mihai Datcu

Reviewed By

Vlad Manilici

2014-10-02

Released By

Dietmar Preuß

© **Restriction of Disclosure:** All rights reserved. No part of this document may be reproduced, in any form or by any means, without permission in written form by the EOLib Project Manager of the German Aerospace Center (DLR).



TABLE OF CONTENTS

1. Introduction	4
1.1 Purpose & Scope	4
1.2 Applicable Documents	4
1.3 Document Structure	4
2. Overview of Image Information Mining	5
3. Active Learning based on SVM	7
3.1 SVM Principles	7
3.2 Active Learning	8
4. Cascade Active Learning for Object Retrieval Algorithm Design	9
4.1 Dataset Module	10
4.2 Learning Module	11
4.3 GUI Module	12
Appendix A References	13

LIST OF FIGURES

Figure 2-1 Overview of the learning system [2]	5
Figure 2-2 User interface of the learning system	6
Figure 4-1 Dataset scheme	10
Figure 4-2 Active learning data scheme	11
Figure 4-3 GUI interface data scheme	12

1. Introduction

1.1 Purpose & Scope

EOLib is an *Image Information Mining* system for *Earth Observation* (EO) data. It processes, extracts and accesses the content of EO products. It stores higher-level abstractions of these products as semantic annotations. EOLib offers information mining services on the original corpus of EO products based on the EO content, associated metadata, semantic annotations and information from external GIS sources.

This document details the Image Information Mining algorithmic design and implementation as used in the EOLib architecture [AD02]. For more details on the algorithmis see also [1].

1.2 Applicable Documents

The following project documents contain provisions which, through reference in this text, become applicable to the extent specified in this document.

[AD01]	EOLib Algorithm Specification	EOLIB-TN-DLR-3300 Issue 1.5, 2013-12-03
[AD02]	EOLib Architecture Concept	EOLIB-TN-DLR-3100 Issue 1.1, 2013-04-04
[AD03]	EOLib Feature Extraction Algorithms	EOLIB-TN-DLR-4400 Issue 0.1, 2014-00-00

1.3 Document Structure

This document is organized as follows:

Chapter 2	presents an overview of the active learning system within EOLib.
Chapter 3	describes the active learning based on SVM (Support Vector Machine).
Chapter 4	defines the implementation of Cascade Active Learning for Object Retrieval.
Appendix A	cites the references to the scientific publications.

2. Overview of Image Information Mining

The *Image Information Mining (IIM)* system is developed for fast indexing and annotation and *Knowledge Discover in Databases (KDD)* exploration of large volume of EO data archive, which works at patch level [2]. Different from a ranking system, this system helps to discover the categories, thus labeling images. The overview of the system is shown in Figure 2, which is composed of primitive feature extraction [AD01], relevance feedback based on active learning, and visualization, besides the *graphical user interface (GUI)*.

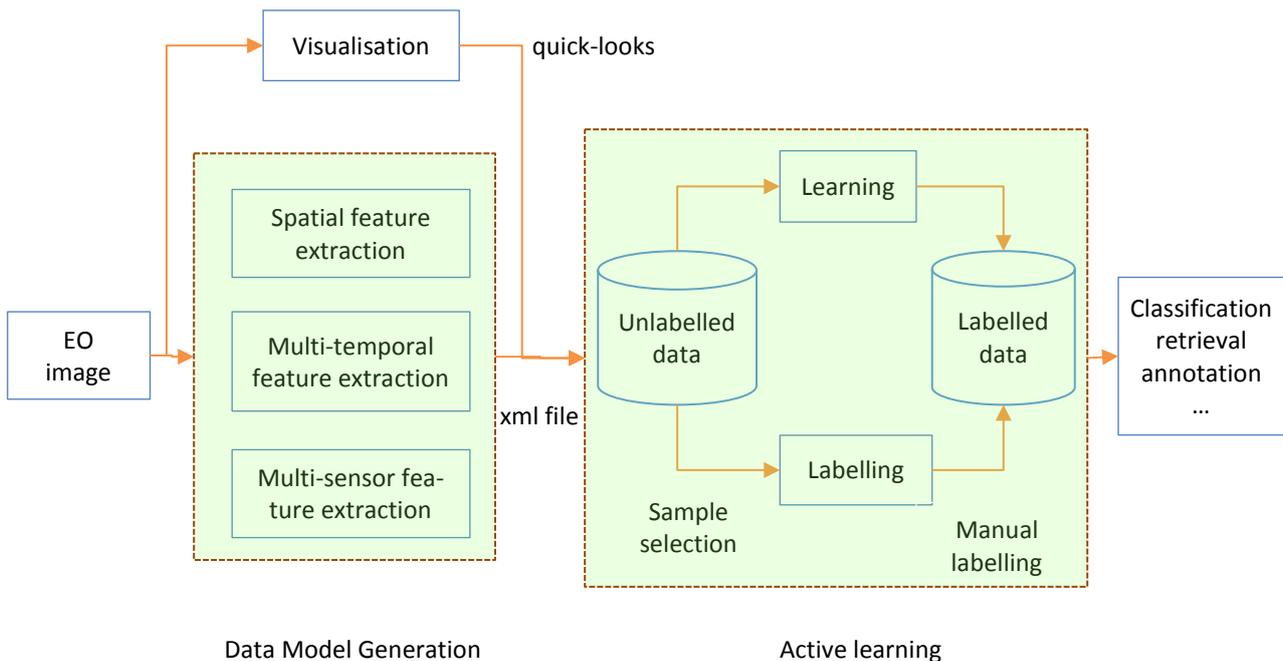


Figure 2-1 Overview of the learning system [2]

After data ingestion, the images that have been imported are tiled into patches and the feature extraction is applied to all the patches. Therefore, feature vectors are used as the representations of the patches. Some randomly selected patches are presented to the users for initialization of the annotation. By selecting the desired target and also some negative training samples, the users could start the interactive learning through the system. The iteration of active learning is repeated by the users until the patches that have been retrieved form a homogeneous category. After that the category characteristics and applicable labels are exported into the database.

We mention here that during the running of DMG module (see [AD03]) each product image is tiled into patches and the extracted features vectors are and stored in a xml file format.

Through the IIM system, different kinds of tasks in EO applications can be done, depending on the input data. The data sources that can be ingested into this system include single image, multi-temporal, and multi-sensor data. Both optical and SAR images can be imported for semantic annotation. Single image annotation is to find and index the relevant patterns in an image. Multi-temporal image annotation can be done separately or jointly. Separate annotation of multi-temporal images is the same as single image annotation but applied to multiple images. By joint multi-temporal annotation, it means to index a certain kind of temporal pattern. Multi-sensor image annotation can also be achieved by fusing individual features from each sensor.

The GUI interface for active learning is shown in Figure 2-2 with its three panels. The top-left panel shows the positive patches that have been retrieved while the bottom-left panel is used to show the patches that are close to the current decision boundary in the feature space. Through the top-left panel, users can visually check the consistency of the training (or classification) for a category. In active learning, the patches close to the current boundary in the feature space determine the current classifier [1]. Thus, for the

sake of improving the decision boundary conveniently, all the patches that play a role in determining the classifier are shown in the bottom-left panel. The right large panel shows the image that is being worked on. Multiple images can be visualized in this panel. Through this panel, users can see the distribution of the retrieved patches and all the training samples. The users can also verify the selected training samples by checking the surrounding context as there is a link between patches in the up left panel and the right panel. Training samples can be selected from any one of the three panels.

The mathematical algorithm used for the active learning is described in the following two chapters.

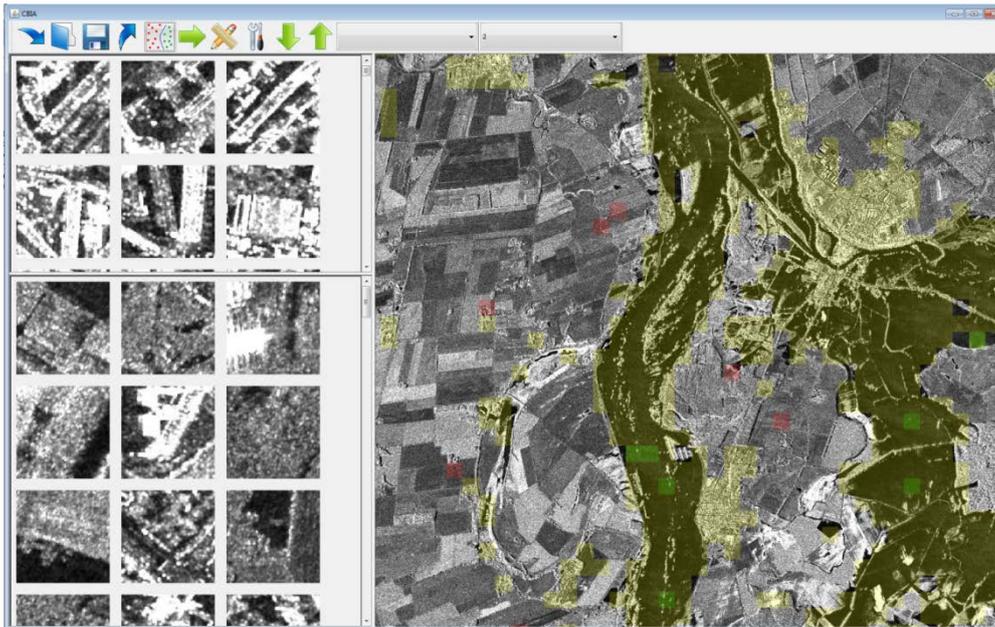


Figure 2-2 User interface of the learning system

The up left panel shows the positive patches that have been retrieved, which are the same as the red patches shown in the overview image on the right hand side. The left down panel shows the patches that are close to the current decision boundary in the feature space, which are important in determining the decision surface. The right large panel shows the image that is being processed.

3. Active Learning based on SVM

Active learning has been widely used in image retrieval [3], and has mainly two components: model learning and sample selection. Model learning is used to train a classifier and sample selection is to choose good training samples that are close to the decision boundary. This calls for a classifier with good performance. In our system, we use Support Vector Machine (SVM), which is briefly recalled in the following.

3.1 SVM Principles

Given N training samples $X = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}_T$, with $x_i \in R^M$, $y_i \in \{-1, +1\}$, the objective of SVM is to find a hyperplane $w^T x + b = 0$ with a normal vector w and bias b that can separate two classes with a maximum margin, subject to the following constraints

$$\begin{aligned} w^T x_i + b &> 0 \text{ for } y_i = 1 \\ w^T x_i + b &< 0 \text{ for } y_i = -1 \end{aligned}$$

The distance $\frac{2}{\|w\|}$ between the two parallel hyperplanes is considered as the margin, which should be maximized such that it has the best generalization ability. With a tolerance of misclassification by introducing slack variables ξ_i , two class linear SVM can then be formulated as a quadratic optimization problem, which is the primal form of soft margin SVM.

$$\begin{aligned} \min_{w, b, \xi} \quad & \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N \xi_i \\ \text{s.t. } & y_i(w^T x_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \quad \forall i = 1, \dots, N \end{aligned}$$

where C is a regularization parameter controlling the trade-off between goodness-of-fit and generalization ability. A higher value of C would penalize the error more, which allows less misclassification, leading to a model better fitting the data but with a risk of overfitting. On the contrary, a smaller C would give an underfitting model, which allows more misclassification.

It is worth to note that the number of variables to be optimized is the same as the dimension of the feature vector. In the case of high dimensional feature vector which is quite common in image processing, the disadvantage of optimizing the primal form would be obvious. Therefore, the dual form is more preferable in the literature, which can be obtained by introducing Lagrange multiplier for each constraint and substituting the primal variable represented using Lagrange multiplier into the primal form.

$$\begin{aligned} Q(\alpha) &= \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i,j=1}^N \alpha_i \alpha_j y_i y_j x_i^T x_j \\ \text{s.t. } & \sum_{i=1}^N \alpha_i y_i = 0, \quad 0 \leq \alpha_i \leq C, \quad i = 1, \dots, N. \end{aligned}$$

From the dual form, we can see that the number of variables to be optimized is the same as the number of training samples and the objective function depends only on the dot product of the data points. These are the main advantages of the dual form over the primal form. Thus, if the dimension of the feature vector is high and only few training samples are available, dual form is more preferable than the primal form.

Although soft margin SVM classify nonlinearly separable data points by allowing misclassified, it cannot manage nonlinear separable feature space essentially. To overcome this drawback, the kernel trick by mapping $\phi: R^N \rightarrow R^M, M \geq N$ the data points to a high dimensional separable space $F = \{\phi(x) | x \in R^N\}$ is introduced based on the fact that the dual form objective functions depends only on the dot product of the feature vector. The only difference is that the dot product between the feature vector should be replaced with the dot product in the high dimensional feature space after mapping. However, it is difficult to identify what feature mapping can separate nonlinear feature space. Nevertheless, the most important achievement in SVM is the dot product after feature mapping is equivalent to a kernel function under some conditions based on Mercer's theorem $K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$. Therefore, the difficult to identify feature mapping function $\phi(x)$ is avoided by a kernel function. The optimization in dual form which depends only

on the Gram matrix with each entry representing the similarity of two feature vectors in terms of a kernel function.

$$Q(\alpha) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i,j=1}^N \alpha_i \alpha_j y_i y_j K(x_i, x_j)$$

$$\text{s.t. } \sum_{i=1}^N \alpha_i y_i = 0, 0 \leq \alpha_i \leq C, i = 1, \dots, N.$$

The corresponding decision function of a test feature vector is

$$f(x) = \sum_{i \in S} \alpha_i y_i K(x_i, x) + b$$

$$K_{\chi^2}(x, y) = \sum_{i=1}^n \frac{(x_i - y_i)^2}{x_i + y_i} \quad K_{HIK}(x, y) = \sum_{i=1}^n \min(x_i, y_i).$$

The complexity of learning a SVM classification depends on the dimension of the Gram matrix, that is the number of training samples. The remaining problem is to choose an efficient kernel function. In most applications, histogram based feature representation of the image is becoming the state-of-the-art method. Empirically, histogram intersection kernel and Chi square kernel perform better than Euclid distance. In addition, they are additive kernels [4], which can be learnt quickly. Chi-square kernel is used in this system.

3.2 Active Learning

In practice, it is usually costly to obtain labels due to limited financial and human resources. The advantage of building a compact yet sufficient training data set has been realized. Starting from a limited number of labeled data, active learning selects the most informative samples to speed up the convergence of accuracy and to reduce the manual effort of labeling. In active learning, it is allowed to query the important instances in the feature space to maximize the accuracy and reduce the time of learning. Actually, many labeled instances in the training data set are not helpful to learn an efficient classifier, instead, they increase the computational complexity. To eliminate those redundant samples and keep only the most informative ones is the main objective of active learning. The two core components in active learning are the selection strategy and model learning, which are repeated until convergence.

The overall framework of active learning is shown in the middle part in Figure 2. At the beginning of the procedure, there are only a few labeled instances are available, a coarse classifier is learnt. After that repeat the iteration of the two components until the classification result is satisfactory. In each iteration, most informative samples from the unlabeled instances in the pool are selected and labeled manually by the oracle to improve the current classifier. These new labeled instances are added to the training set and a new classifier is learnt again. There are many different active learning paradigms, such as membership query learning, stream-based active learning, and pool-based active learning [5]. In this, we focus on pool-based active learning, where a large pool of unlabeled data and only a small set of labeled data are available. In each iteration, queries are selected from the pool based on an informative measure. Thus, sample selection strategy plays a crucial role in active learning. In [6] are review various sample selection strategies.

4. Cascade Active Learning for Object Retrieval Algorithm Design

Cascade Active Learning for Object Retrieval (CALOR) is the algorithm considered for KDD and includes the SVM component that was presented in previous chapter.

CALOR [7] is based on a hierarchical top-down processing scheme for object retrieval in high volume and high resolution optical satellite repositories. It learns via a multistage active learning process a cascade of classifiers working each at a certain scale on a patch-based representation of images. At each stage of the hierarchy we seek to eliminate large parts of images considered as non-relevant, the purpose being to set the focus at the finest scales on more promising and as spatially limited as possible areas. The scheme is based on the fact that by reducing the size of the analysis window (i.e. the size of the patch), to better capture the properties of the targeted object. The cascaded hierarchy is introduced to compensate for the extra computational burden incurred by diminishing the size of the patch, which causes an explosion of the number of patches to process. [8] proposed a cascade active learning strategy to build a classifier at each level of the hierarchy and we provide a new Multiple Instance Learning Algorithm to propagate automatically the training examples from one level of the hierarchy to the other.

The entire system is designed following a standard object-oriented procedure. The theoretical background and the results are presented in [1]. The system implementation has three main modules: Graphical User Interface (GUI), Learning model and dataset model, as shown in Figure 4, Figure 5, and Figure 6. Module Design

4.1 Dataset Module

The dataset module manages the output of the *Data Model Generation* datasets. It has several main classes, such as dataset, category, patch, scene and annotations. The relationships between these classes are shown in Figure 4. Physically, patch corresponds to an image patch and scene corresponds to a TerraSAR-X product. Category denotes a semantic class obtained after annotation. In this design, class dataset is a singleton pattern since there is only one dataset ingested into the system at any time. This module provides the features and quick looks for the learning module and obtains the semantics after learning.

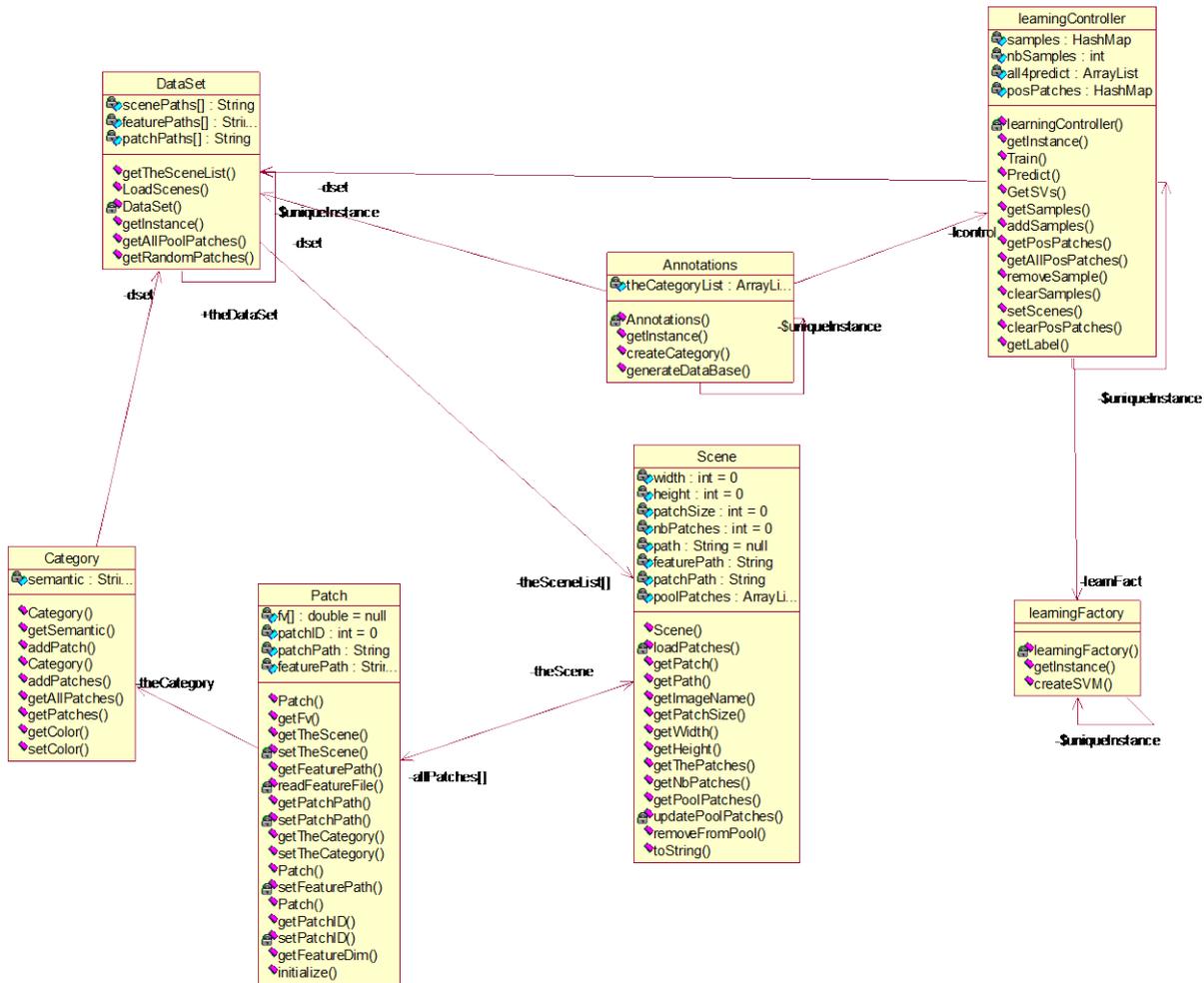


Figure 4-1 Dataset scheme

4.2 Learning Module

The learning module provides all the functions about learning and it does not manage anything about dataset. The feature files and quick look images are taken from the dataset module. The UML diagram of this module is shown in Figure 5. In this module, the learning algorithms are mainly Support Vector Machines (SVMs). For the sake of speeding up, this module includes both linear and nonlinear SVM [9]. In most cases, linear SVM is much faster than nonlinear SVM, but its accuracy can be lower. The quadratic optimization algorithm involved in nonlinear SVM is solved by the class QuadSolver, which implements a Sequential Minimal Optimization algorithm (SMO). The kernel functions adopted in this system are Chi-square and histogram intersection kernel (HIK) because they are additive kernels and perform very well for histogram-kind of features.

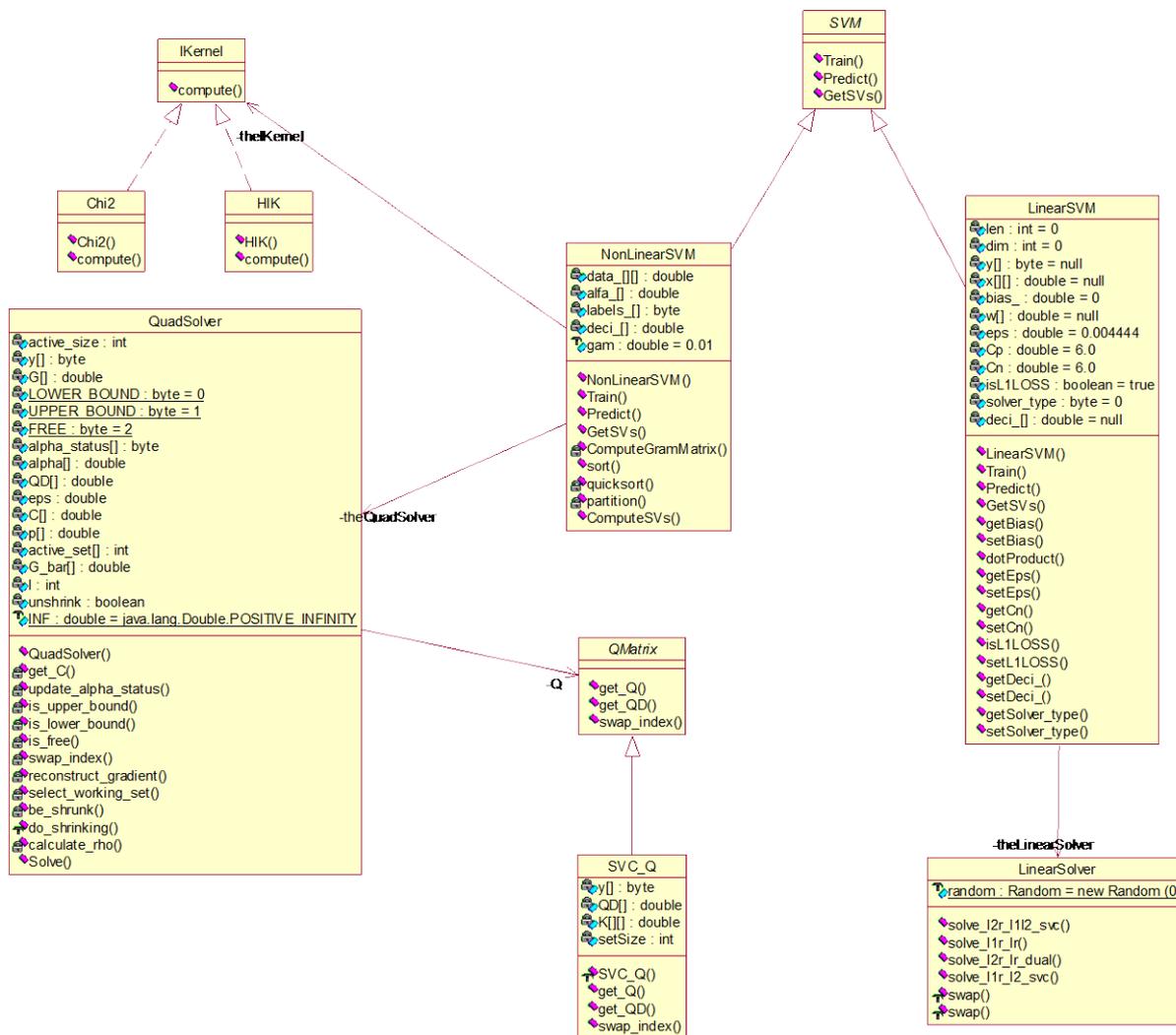


Figure 4-2 Active learning data scheme

4.3 GUI Module

The GUI module manages all the aspects about interface and user interaction. It provides the panels for patch browsing and image visualization. The UML diagram is shown in Figure 6. User interaction is performed through selecting on relevant panels. These selected samples are feed into the learning module.

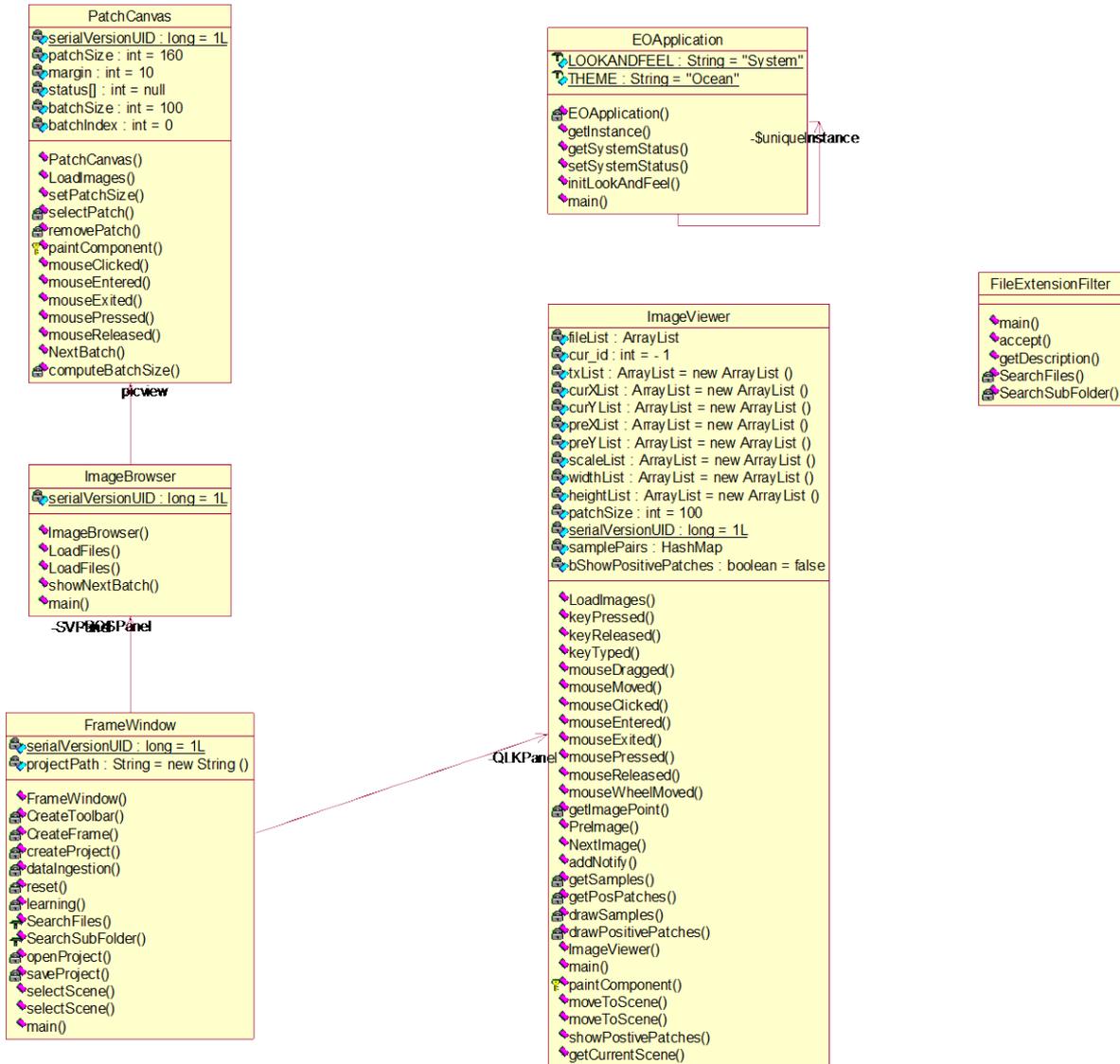


Figure 4-3 GUI interface data scheme

Appendix A References

- [1] P. Blanchart, M. Ferecatu, S. Cui, M. Datcu (2014). Pattern Retrieval in Large Image Databases using Multiscale Coarse-to-Fine Cascaded Active Learning. *IEEE Journal of Selected Topics in Applied Earth Observation and Remote Sensing*, 7 (4), pp. 1127-1141.
- [2] S. Cui, C. Dumitru, and M. Datcu (2014). Semantic annotation in Earth observation based on active learning. *International Journal of Image and Data Fusion*, 5, pp. 152-174.
- [3] M. Costache, H. Maitre, and M. Datcu (2006). Categorization based relevance feedback search engine for earth observation images repositories. *Geoscience and Remote Sensing Symposium (IGARSS)*, 2006 IEEE International, pp. 13-16.
- [4] S. Maji, A. C. Berg, and J. Malik (2013). Efficient classification for additive kernel SVMs. *IEEE Trans. Pattern Anal. Mach. Intell.* 35 (1), pp. 66-77.
- [5] B. Settles. (2009). Active learning literature survey. *Computer Sciences Technical Report 1648*, University of Wisconsin-Madison.
- [6] S. Tong, and D. Koller (2002). Support vector machine active learning with applications to text classification. *J. Mach. Learn. Res.* 2 (22), pp. 45-66.
- [7] P. Blanchart, M. Ferecatu, and M. Datcu, (2011). Cascaded Active Learning for Object Retrieval using Multiscale Coarse to Fine Analysis. *IEEE Conference on Image Processing (ICIP)*. Brussels, Belgium.
- [8] P. Blanchart. (2011). *Fast learning methods adapted to the user specificities: application to earth observation image information mining*. Paris: École Doctorale d'Informatique, Télécommunications et Électronique de Paris & Télécom ParisTech.
- [9] I. Steinwart and A. Christmann (2008). *Support Vector Machines*. New York, NY, USA: Springer.