Context Inclusive Function Evaluation: A Case Study with EM-Based Multi-Scale Multi-Granular Image Classification

Vijay Gandhi, James Kang, Shashi Shekhar, Junchang Ju, Eric D. Kolaczyk, and Sucharita Gopal

July 30, 2008

Context Inclusive Function Evaluation: A Case Study with EM-Based Multi-Scale Multi-Granular Image Classification

Vijay Gandhi, James M. Kang*, Shashi Shekhar
University of Minnesota [gandhi, jkang, shekhar]@cs.umn.edu

Junchang Ju, Eric D. Kolaczyk, Sucharita Gopal
Boston University [junchang, kolaczyk, suchi]@bu.edu

Abstract

Many statistical queries such as maximum likelihood estimation involve finding the best candidate model given a set of candidate models and a quality estimation function. This problem is common in important applications like land-use classification at multiple spatial resolutions from remote sensing raster data. Such a problem is computationally challenging due to the significant computation cost to evaluate the quality estimation function for each candidate model. For example, a recently proposed method of multi-iscale, multi-granular classification has high computational overhead of function evaluation for various candidate models independently before comparison. In contrast, we propose an upper bound based context-inclusive approach that reduces computational overhead based on the context, i.e. the value of the quality estimation function for the best candidate model so far. We also prove that an upper bound exists for each candidate model and the proposed algorithm is correct. Experimental results using land-use classification at multiple spatial resolutions from satellite imagery show that the proposed approach reduces the computational cost significantly.

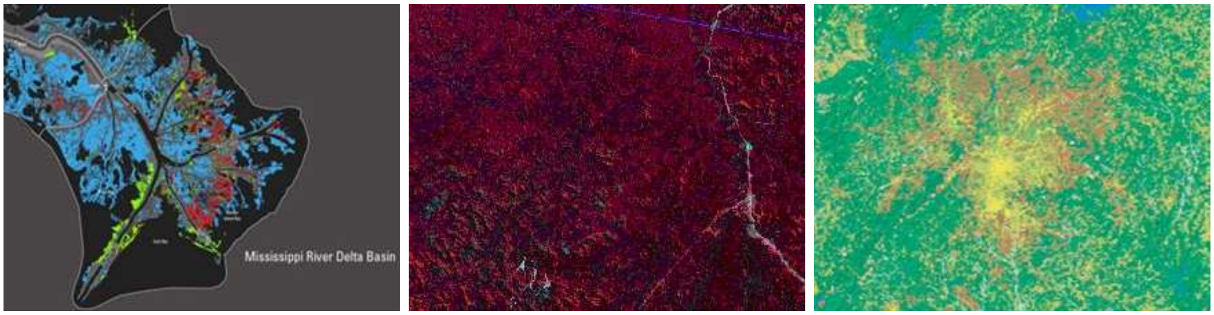
1 Introduction

We are interested in a probabilistic statistical query to find the preeminent candidate model from a set of candidate models using a quality estimation function. We refer to such a problem as the *best candidate model problem*. Formally, it can be stated as follows: Given a set of candidate models and a function to evaluate the quality of each candidate model, the goal is to find the best candidate model probabilistically. The evaluation of this measure is generally very expensive and thus minimizing the computation time is a key objective.

One important example of the *best candidate model problem* is in classification of a spectral image, obtained from a satellite, with domain-specific labels to produce a *thematic map*. *Thematic maps* are widely used in applications including agricultural monitoring, land cover change analysis, and environmental assessment. Examples in land cover change can be seen in Figure 1, where 217 square miles of Louisiana's coastal lands were transformed to water after hurricanes Katrina and Rita (Figure 1a); the deforestation in Brazil causing the loss of 150,000 sq. km. of forest between May 2000 and August 2006 (Figure 1b); and urban sprawl in Atlanta, GA between 1976 and 1992 (Figure 1c).

Image classification at multiple spatial resolutions is an important application of spatial data mining. For example, NASA's Earth observation systems obtain a spectral image of land-use, which is then classified at multiple resolutions. The *best candidate model problem* to find the best classification label can be considered as a parameter estimation problem. Since estimating parameters at a spatial region is an important function in spatial data mining, the *best candidate model problem* may be a sub-class of spatial data mining.

*Corresponding Author



(a) Mississippi River Delta, Lousiana (Red rep. land loss between 2004 and 2005. Courtesy: USGS)

(b) Deforestation in Brazil. (Courtesy: Global Change Program, University of Minnesota)

(c) Urban Sprawl in Atlanta, GA. (Red indicates expansion between 1976 and 1992)

Figure 1: Land Cover Change Examples. (Best viewed in color)

Numerous studies in remote sensing have been done for multi-resolution land-use classification (e.g., [5, 10, 12]). See [16] for a detailed discussion on various methods for multi-resolution classification. A statistical method to classify an image at varying spatial and categorical resolutions was proposed in [9]. This approach is context-exclusive, based on using a query tree to identify each candidate model independently. The maximum likelihood is used as a set operator among quality measures for each candidate model, thus making it necessary to analyze all candidate models together. The result is very high computation costs to identify the preeminent candidate model.

Figures 2 and 3 gives an example of a multi-scale, multi-granular image classification based on land usage. Figure 2 gives the input that includes a set of domain-specific labels (also called classes) logically grouped as a hierarchy (Figure 2a), and the values of each specific class: $conifer$, *hardwood*, *brush*, and *grass* (Figures 2c-f). The values of a specific class are derived from a synthetic remotely-sensed satellite image [9] (Figure 2b), that has been provided in this paper for completeness. The goal is to classify each pixel in the satellite image to one of the labels based on a quality measure called *likelihood*. The *likelihood* measure is calculated for non-specific classes using Expectation Maximization (EM) [1] which is computationally expensive because of the large number of iterations required until convergence. Also, multiple scales are defined implicitly in powers of 2 (i.e., $2 \times 2, 4 \times 4, \ldots, 2^{n-1} \times 2^{n-1}$, where n is the size of the image). An example output in shown in Figure 3 having a set of classified images at scales $1 \times 1, 2 \times 2, 4 \times 4$, and 64×64.

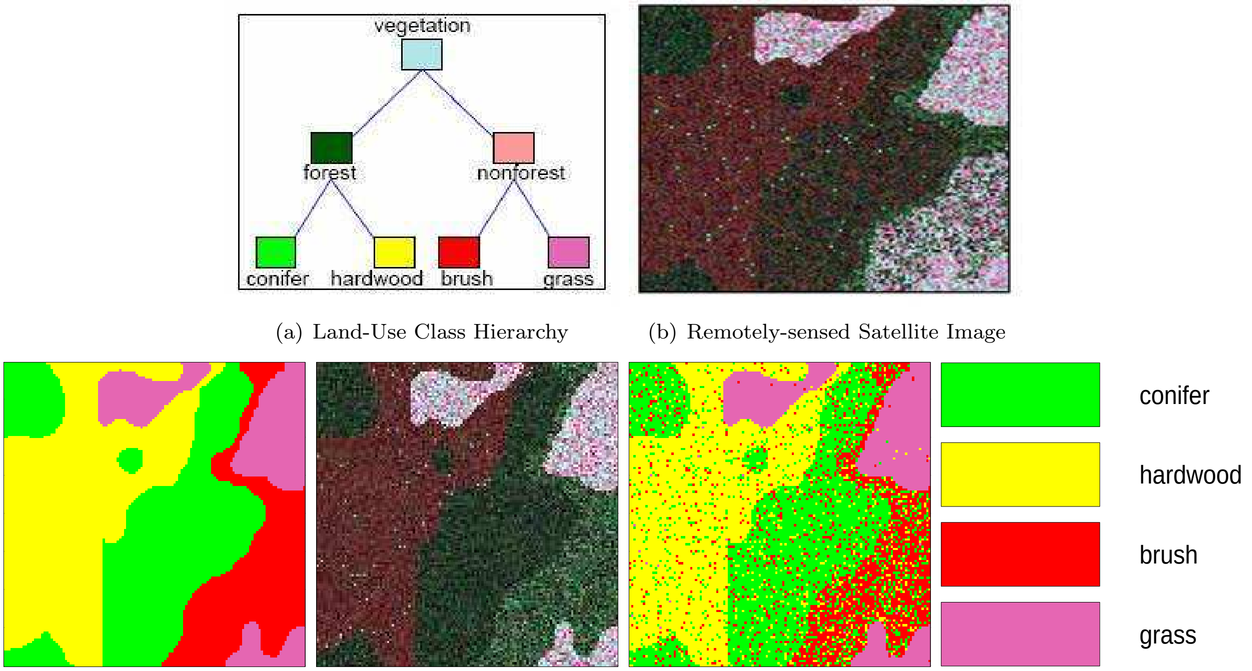
Calculating the *likelihood* of each candidate model makes the problem computationally expensive. For instance, the approach proposed by [9], takes about 7 hours of computation time to classify an image of size 512 x 512 pixels with 12 labels at varying *spatial resolutions*. About 80% of the total computation time is consumed to find the quality measure for each candidate model. Thus, as the image size grows the computation time increases, which makes this problem challenging.

The focus of our work is to develop a computational efficient EM based multi-scale multi-granular (MSMG) method (see [9] for details on MSMG). In our previous paper [2], we proposed two heuristics to reduce the computation time for the evaluation of candidate models. The first heuristic, called Limiting Factor (LF), is based on the precision of convergence for each candidate model. As LF decreases, the computation time is reduced due to faster convergence of each candidate model. The second heuristic, called context-inclusive heuristic (CIH), simultaneously considers information from all models during evaluation rather than independently as in the context-exclusive (CE) approach. Rather than executing each model till convergence and then ranks the models as in CE, CIH evaluates all models together as it executes and exits as soon as a single model is dominant over all the others.

However, CIH may not obtain the same exact results as in CE due to the fact that each model may not fully converge. Thus, we propose an upper bound based context-inclusive approach (UBCI) that utilizes a filter and refine technique for each model. We provide an upper bound for each model and use this information to obtain the same exact results as CE while still obtaining a significant savings in computation time than the CE approach.

Initial results were reported in [2], this paper makes additional new contributions:

1. We propose a novel Upper Bound based Context-Inclusive approach that finds correct classifications while achieving a significant reduction in computation time (Section 3.2).



(a) Land-Use Class Hierarchy (b) Remotely-sensed Satellite Image

(c) 'true' class labels (d) color composite (e) per-pixel classification (f) legend

Figure 2: Example Input: Class Hierarchy, Satellite Image, and values for specific classes [8] (Best Viewed in Color)

2. We prove than an upper bound exists for each model is our best candidate model problem (Section 4.1).
3. A proof of correctness is provided for our proposed approach (Section 4).
4. Experimental evaluation of synthetic and real datasets is given in terms of the number of iterations in EM and the computation time for our proposed method.

In addition, we have made several revisions to improve its readability. For example, we provide a new section discussing Expectation Maximization (EM), its application to our problem, and an example execution trace in Section 2.

Scope. The following topics are out of scope in this paper: (i) the choice of using an optimal EM and statistical models for Multi-Scale Multi-Granular (MSMG) classification, (ii) the actual accuracy of the context exclusive approach, and (iii) a comparison between MSMG and traditional approaches. Interested readers are encouraged to see [6, 7] for more details.

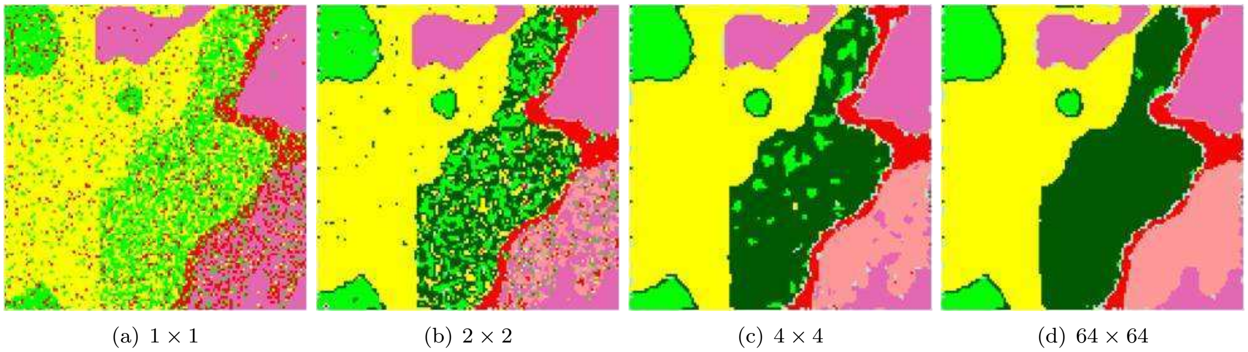
Outline. The rest of the paper is organized as follows: A background on the EM algorithm and its application to our problem is described in Section 2. Section 3 gives a detailed overview of our approach along with the major differences with previous work. As part of our proposed approaches, we prove an upper bound exists in each candidate model (Section 4). Experimental results to compare the previous and the proposed approach are given in Section 5. Finally, Section 6 concludes this paper with a discussion and future work.

2 Background

This section presents a general overview of Multi-Scale Multi-Granular Image classification and its application to the best candidate problem. Also, an execution trace example is shown.

2.1 EM-Based Multi-Scale Multi-Granular Image Classification

Satellite remote sensing provides timely and continuous coverage of the earth's surface and has proven to be extremely important in making thematic maps such as land cover maps. Such remote sensing data is available now



(a) 1 × 1 (b) 2 × 2 (c) 4 × 4 (d) 64 × 64

Figure 3: Example Output: Image Classification at Multiple Scales (Please refer to Figure 2a for the legend)

at a variety of spatial scales as part of the Earth Observing System (EOS) enterprise undertaken by NASA. This explosion of satellite data is leading to a paradigm shift in the way classification is done. Land cover classification in remote sensing is traditionally carried out at a single spatial scale and a single categorical scale and increasingly, the need to integrate data at multiple spatial scales, is leading to a new emphasis in classification that is both multiscale, in the sense of being able to incorporate information across multiple spatial scales in constructing a map and also dmultigranularin the sense that class labels of different granularity, or categorical scale, similarly co-exist in the same map. As discussed in Ju et al. [7], multiscale and multigranular framework may improve the accuracy of pixel-wise classification as well as impact the validation of coarse-scale land cover classification, provide an adaptive choice of scale in correspondence with local complexities in the image as well address a problem relating to cartographic generalization, where the objective is to best represent selected classes of features at different map scales. Our framework provides a statistically grounded procedure for producting landcover maps that are potentially more intuitive and visually appealing than those created using monogranular methods

In this section, we present a mathematical representation of our previous method [9]. Please use Table 1 as a reference for the list of notations used in this section. Suppose we have digitized a spatial region into an image of n^2 pixels i.e., $\{I_{i_1,i_2}\}_{i_1,i_2=1}^{n}$. For each pixel I_{i_1,i_2}, there is a vector-valued observation x_{i_1,i_2}. The landcover content of this pixel is denoted as c_{i_1,i_2}, and is assumed to be a member of some set of specific classes $\mathcal{C} = \{1, \ldots, C\}$.

A typical mono-granular probability model for a measurement x, at a pixel I, will specify the likelihood through a class-specific density, say $g(x|c)$, for x conditional on the underlying truth c. But the multiscale/multigranular probability model [9] allows the user to assign a less-specific class to an entire region R containing I and certain surrounding pixels. The choice of possible regions R is restricted to those allowed under a quad-tree decomposition of the image region. Less-specific classes are constrained to belong to a subset S, of the specific classes in $\mathcal{C}$. Such subsets are pre-defined, based on prior knowledge of the image domain, and restricted in number.

The likelihood for the multiscale/multigranular probability model is specified using a mixture model i.e.,

$$f(x|S) = \sum_{c \in S} \pi_c g(x|c) \quad . \tag{1}$$

The class-specific densities $g(x|c)$ are assumed known i.e., they are input for the overall algorithm. So the unknown quantities in the model are (i) the choice of regions R, (ii) the choice of subsets S, for each region R, and (iii) the choice of relevant weights π_c, for each class $c \in S$. These unknown quantities must be inferred from data.

A maximum likelihood approach is natural here, using the pixel measurements $\{x_{i_1,i_2}\}$. Note that since the regions R can contain more than one pixel, by design, selection of the model parameters for any one pixel involves an optimization incorporating all pixels. And, since this is a very rich model class, some sort of penalty is adviseable to discourage over-fitting. A complexity-penalized maximum likelihood method for model inference was proposed in [9].

Specifically, let $\mathcal{M} = (\mathcal{R}, \mathcal{S}, \Pi)$ be the set of regions R, subsets $S = S(R)$ of class labels c, and weights $\pi_c = \pi_c(S)$ under a given MSMG model for the image region. The optimal model $\hat{\mathcal{M}}$ is chosen as

$$\hat{\mathcal{M}} \equiv \arg\max_{\mathcal{M}} \; \{ \, \ell \, (\mathbf{x} \, | \, \mathcal{M}) - \; \mathrm{pen} \, (\mathcal{M}) \, \} \quad , \tag{2}$$

Notation	Definition	Notation	Definition
n	Number of Pixels	R	Spatial Region
I_{i_1,i_2}	Pixel in an image	π_c	Weights for class c
x_{i_1,i_2}	Vector valued observation	$\mathcal{M}$	MSMG Model
C	Set of specific classes	$\hat{\mathcal{M}}$	Optimal MSMG Model
C_i	class i	$\mathcal{S}$	Collections of possible subsets
$g(x\|c)$	likelihood value for x given c	m	Number of specific classes
$L(\mathbf{x}\,\|\,\mathcal{M})$	Likelihood	$\ell(\mathbf{x}\,\|\,\mathcal{M})$	Log-Likelihood

Table 1: Table of Notations

where $\ell(\mathbf{x}\,|\,\mathcal{M})$ is the log-likelihood of the data under model $\mathcal{M}$ and $\text{pen}(\mathcal{M})$ is a penalty function. Here $\mathbf{x} = \{x_{i_1,i_2}\}$ represents the full set of observations.

The penalty in (2) is essentially the code-length for describing the model $\mathcal{M}$, a precise definition of which can be found in [9], the likelihood is of the form

$$L(\mathbf{x}\,|\,\mathcal{M}) = \prod_{i_1,i_2} f(x_{i_1,i_2}\,|\,S_{i_1,i_2}) \ \ , \tag{3}$$

and the log-likelihood is of the form

$$\ell(\mathbf{x}\,|\,\mathcal{M}) = \sum_{i_1,i_2} \log f(x_{i_1,i_2}\,|\,S_{i_1,i_2}) \ \ , \tag{4}$$

where the quantities $f(x_{i_,i_2}\,|\,S_{i_1,i_2})$ are mixture densities like that in (1). Now if the densities $f(x|S)$ in (2) were known, then optimization would reduce to a search over all partitions of the image into regions R, and assignment of subsets S of categories to each region R. Give the use of a quad-tree structure in defining the regions R, this would result in an algorithm of $O(n^2|\mathcal{S}|)$ complexity, where $\mathcal{S}$ is the collection of all possible subsets S allowed by the user.

However, the mixture weights in the densities $f(x|S)$ are unknown and must be fit for each candidate region R and each candidate subset S. Usually estimation of mixture weights through maximum likelihood is done using the EM algorithm. Thus, the optimization in (2) nominally requires $O(n^2|\mathcal{S}|)$ EM algorithms to be run. This requirement represents the major bottleneck in optimizing computational performance for the MSMG methodology and is this aspect that this paper seeks to improve upon.

2.2 Execution Trace

There are many implementations of the Expectation Maximization algorithm [1]. Algorithm 1 presents the Expectation Maximization algorithm that is used in our problem domain as described in [9]. First, we present the algorithm to discuss the computation aspect but not the correctness. Second, we give an example of this Algorithm using a two class problem. The input to Algorithm 1 is the Non-Specific class (e.g. $forest$) to determine the maximum likelihood of its specific class (e.g. $conifer$ and $hardwood$) proportions π_1 and π_2 respectively, the likelihoods of each of the specific classes, and the spatial region (e.g. a quad of size 2 x 2 pixels).

In general, there are five major steps in a single iteration of Algorithm 1. First, the proportions are set to their current values. If this is the first iteration, then the proportions are initialized such that each has an equal weight totaling to one (e.g., for two specific classes, the proportions are $\pi_1 = 0.5$ and $\pi_2 = 0.5$. Otherwise, the proportions are obtained in the previous iteration. Second, the product of the likelihood at each pixel for a specific class and its proportion is set to $h(x_{i_1,i_2}|C_k)$. For example, for two classes, the following would be calculated:

$$\begin{array}{rcl} h(x_{i_1,i_2}|C_1) & = & g(x_{i_1,i_2}|C_1) * \pi_1 \\ h(x_{i_1,i_2}|C_2) & = & g(x_{i_1,i_2}|C_2) * \pi_2 \end{array} \tag{5}$$

The third step is to take the sum of all likelihood values from all the specific classes. For two classes, the following would be used:

$$t(x_{i_1,i_2}) \quad = \quad h(x_{i_1,i_2}|C_1) + h(x_{i_1,i_2}|C_2) \tag{6}$$

Fourth, for each specific class, the values obtained in Step 2, $h(x_{i_1,i_2}|C_k)$, is divided upon the values obtained in step 3, $g^{'}$, and is set to $j(x_{i_1,i_2}|C_k)$. For example, in two classes:

$$\begin{aligned} j(x_{i_1,i_2}|C_1) &= \tfrac{h(x_{i_1,i_2}|C_1)}{t(x_{i_1,i_2})} \\ j(x_{i_1,i_2}|C_2) &= \tfrac{h(x_{i_1,i_2}|C_2)}{t(x_{i_1,i_2})} \end{aligned} \tag{7}$$

Finally, the fifth step averages the values in $j(x_{i_1,i_2}|C_k)$ for each specific class k and the new proportion π_k. For example, in two classes the proportions are found by:

$$\begin{aligned} \pi_1 &= avg_{x_{i_1,i_2}}(j(x_{i_1,i_2}|C_1)) \\ \pi_2 &= avg_{x_{i_1,i_2}}(j(x_{i_1,i_2}|C_2)) \end{aligned} \tag{8}$$

This process is repeated until the desired accuracy of the proportions (i.e., difference between the current and previous proportions) is acquired. Accuracy is determined by the amount of change in the likelihood value based on the proportion size between iterations.

Algorithm 1 EM-based MSMG Image Classification

```
Function EM(class Non-Specific)
Initialize the proportions as equal weights
repeat
    Step 1 Set the proportion of each corresponding specific class in the spatial region
    Step 2 Multiply likelihood at each pixel in the spatial region by corresponding specific class proportion
    Step 3 Add the likelihood at corresponding pixel
    Step 4 Divide the value in Step 2 by value in Step 3 at corresponding pixel
    Step 5 Average the likelihood in the spatial region for each specific class and consider these to be new proportions
until the required accuracy is achieved
return sum of the product of proportions and likelihood Maximum Likelihood
```

To give an example, consider the use of the Expectation Maximization algorithm for a non-specific class $forest$, calculated for a quad of size 2 x 2 pixels. The given likelihood values of the specific classes, Conifer (C_1) and Hardwood (C_2), corresponding to $forest$ are represented by vectors in Table 2.

(a) Conifer

0.2	0.2
0.9	0.9

(b) Hardwood

0.8	0.8
0.1	0.1

Table 2: Likelihood values (in 10^{-4}) for specific classes of Forest in a 2x2 region

Based on Algorithm 1, there are five major steps in a single iteration. First, the proportions are initialized to equal weights since this is the first iteration, $\pi_1 = 0.5$ and $\pi_2 = 0.5$. Second, based on Eq. 5, $h(x_{i_1,i_2}|C_k)$, and using the likelihood values in Table 2 and the initial proportion values (Table 3), $h(x_{i_1,i_2}|C_k)$ is obtained in Table 3.

(a) $h(x_{i_1,i_2}|C_1)$

0.1	0.1
0.45	0.45

(b) $h(x_{i_1,i_2}|C_2)$

0.4	0.4
0.05	0.05

Table 3: Step 2 Results from Algorithm 1

The third step uses Eq. 6 to find the total value between all specific classes. In our running example, Table 4 gives the results for $g^{'}$ by summing the values obtained in the second step.

The fourth step uses the values from steps 2 and 3 within Eq. 7 and the values from are 2 class example is given in Table 5.

Finally, the new proportions for each specific class can be created by taking the average value in $j(x_{i_1,i_2}|C_k)$. In our example, the new proportions are $\pi_1 = 0.55$ and $\pi_2 = 0.45$. The process will continue to iterate until the 16th iteration where the final converging proportions are $\pi_1 = 0.6042$ and $\pi_2 = 0.3958$. The log-likelihood for $forest$

0.5	0.5
0.5	0.5

Table 4: $g^{'}$, Step 3 Results from Algorithm 1

(a) $j(x_{i_1,i_2}|C_1)$

0.2	0.2
0.9	0.9

(b) $j(x_{i_1,i_2}|C_2)$

0.8	0.8
0.1	0.1

Table 5: Step 4 Results from Algorithm 1

can be calculated using Eq. 4 which is 1.17162. The likelihoods for the specific classes $conifer$ and *hardwood* can be obtained by simply summing the values in Table 2 which is 2.2 and 1.8 respectively. Thus, the class having the maximum likelihood from $forest$, $conifer$, and *hardwood* is $confier$ with a log-likelihood value of 2.2 and is assigned to the given 2x2 region.

3 Approaches

In this section, we present two approaches to address the *best candidate model problem.* The first approach is based on a previous approach [9] which we call context exclusive since each candidate model is evaluated independently (Section 3.1). The second approach is our proposed upper bound based context-inclusive method that uses a filter and refine technique to prune models based on their upper bound while evaluating each simultaneously (Section 3.2).

3.1 Context-Exclusive Approach

Algorithm 2 presents the pseudo code of the context-exclusive approach from [9]. The input to Algorithm 2 is the set of non-specific class candidates and the output is the arg max candidate in the *candidate set* and its corresponding likelihood value (i.e. maximum value in the Set of Candidate and Likelihood values *SCL*). The main objective in Algorithm 2 is to obtain the candidate classification for a spatial region. Each candidate model in the *candidate set* is analyzed independently to obtain its likelihood for a spatial region by executing EM from Algorithm 1 (Line 4 in Algorithm 2). The candidate model containing the maximum or largest likelihood of all candidates in *Cand* is declared the best candidate model (*arg max*) for a spatial region (Line 6 in Algorithm 2).

Algorithm 2 Context-Exclusive Approach

```
1: Function contextExclusive(candidate set)
2: Set of Candidate and Likelihood Values SCL ← {(∅, ∅)}
3: for each candidate $c \in$ candidate set do
4:     SCL ← SCL ∪ ($c$, EM($c$))
5: end for
6: return arg max(SCL), MAX(SCL)
```

To illustrate the context exclusive approach using an example, consider the hierarchy in Figure 2a and the EM values in Figure 4. Initially, the specific class with the highest log-likelihood is chosen. Then, each non-specific class is analyzed in EM until convergence. The number of EM iterations for $Vegetation$, $Forest$, and *Non-Forest* is 46, 34, and 3 respectively. Then, the class with the highest log-likelihood value among all of the classes (specific and non-specific) is chosen. In this example and as shown in Figure 4, *Non-Forest* has the highest value and is assigned to the given region. It is important to note and explained further in the next section that using this example, the CE approach took 83 EM iterations to find the answer.

3.2 Upper Bound based Context-Inclusive Approach

Our proposed approach evaluates each model together to obtain a correct candidate model, a relationship which we refer to as context-inclusive using an upper bound. Essentially, the upper bound allows for pruning of non-specific class models (e.g. Forest) which allows for a significant reduction in computation time while achieving correct results. Using remote sensing terminology, each candidate model tuple represents a classification consisting of either a *specific* (i.e., $conifer$ or *hardwood*) or a *general* (i.e., $forest$) class (see Figure 2a). The quality measure for a *specific* class is a single value whereas a *general* class consists of several proportions of *specific* classes. For example, a *general* class of type $forest$ may have several proportions of $conifer$ and *hardwood* trees. A computationally very expensive function is used (i.e., EM) to identify the likelihood value for each candidate model. The main objective is to find the maximum likelihood value or best candidate model to represent a land-use classification.

Algorithm 3 Upper Bound based Context-Inclusive Approach

```
 1: Function UpperBoundcontextInclusive(Candidate Set)
 2: Select c_cand from the candidate set
 3: ML_cand ← EM(c_cand)
 4: BestCand ← c_cand
 5: for each $c \in$ the candidate set AND $c \neq$ c_cand do
 6:     ThUpper ← Upper Bound for the candidate $c$
 7:     if ThUpper < ML_cand then
 8:         $c$ is pruned from the candidate set
 9:     else
10:         c_ML ← EM(c)
11:         if c_ML > ML_cand then
12:             BestCand ← $c$
13:             ML_cand ← c_ML
14:         end if
15:     end if
16: end for
17: return  BestCand, ML_cand
```

Algorithm 3 presents the upper bound based context inclusive approach. An upper bound is calculated for each non-specific candidate to ensure that the correct candidates are pruned from the *candidate set* (Theorem 1). By determining the upper bound on the likelihood value of a non-specific class (Line 2 of Algorithm 3), it is possible to prune candidates whose upper bound is lower than the current best likelihood value. This ensures that those candidates that have an upper bound less than the current best candidate can not be the best candidate and are then pruned (Lines 7-9 of Algorithm 4). For all other candidates, their respective maximum likelihoods are found through the EM algorithm and compared against the current best candidate (Lines 11-15 of Algorithm 4). Finally, the best candidate having the largest or maximum likelihood of all other candidates is returned (Line 18 of Algorithm 4).

To illustrate with an example, consider the hierarchy defined in Figure 2a and the actual likelihood and upper bound values in Figure 5. Initially the likelihood values are calculated for the specific classes ($conifer$, *hardwood*, *brush*, *grass*) and the upper bounds for the non-specific classes ($forest$, *non-*$forest$, *vegetation*). Then, the class (both specific and non-specific) with the highest value is chosen. Figure 5 shows that the upper bound for *non-*$forest$ is the largest. EM is executed for the *non-*$forest$ class until convergence (3 iterations) which obtains a log-likelihood value of approximately -50.2. Any non-specific class having an upper bound less than -50.2 can be pruned out because its EM converging value cannot be higher. Thus, both $forest$ and *vegetation* can be pruned out, leaving $non-forest$ as the dominant class and is assigned to the given region. Notice that using this example, the Upper Bound based Context-Inclusive approach can obtain the same answer as in the Context Exclusive (CE) approach in only three iterations rather than 83 in CE.

4 Analytical Evaluation

This section presents an analysis on obtaining and using an upper bound for the multi-scale multi-granular image classification. First, we prove an upper bound (Lemma 1) for the candidate models in our problem. Second,

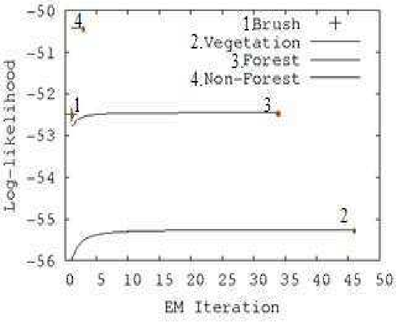


Figure 4: Context Exclusive Approach

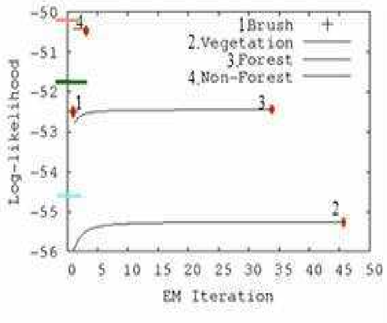


Figure 5: Upper Bound based Context Inclusive Approach. The pink, green, and blue lines is the upper bound for Non-Forest, Forest, and Vegetation respectively. (Note: Best Viewed in Color)

we present the proof of correctness for the Upper Bound based Context Inclusive approach (Theorem 1).

4.1 Upper Bound

Lemma 1 *The pixel-wise sum of the* $log(\max_{c\in S} g(x|c))$ *is an upper bound for the log-likelihood function* $\ell((\mathbf{x}\,|\,\mathcal{M})$*. Formally,*

$$\ell((\mathbf{x}\,|\,\mathcal{M}) \quad \leq \quad \textstyle\sum_{i_1,i_2} \log b_{x_{i_1,i_2},S_{i_1,i_2}} \tag{9}$$

where $b_{x,S} = max_{c\in S} g(x|c)$*.*

Proof Based on Eq. 1, note that

$$f(x|S) = \sum_{c\in S} \pi_c \; g(\mathbf{x}\,|\,c) \leq \max_{c\in S} g(\mathbf{x}\,|\,c) \enspace . \tag{10}$$

Since $\sum_{c\in S} \pi_c \; g(\mathbf{x}\,|\,c)$ is a convex combination [15] of densities $g(\mathbf{x}\,|\,c)$.
Let us revisit the log-likelihood (Eq. 4)

$$\ell(\mathbf{x}\,|\,\mathcal{M}) = \sum_{i_1,i_2} \log f(x_{i_1,i_2}\,|\,S_{i_1,i_2}) \enspace ,$$

and rewrite the right hand side using Eq. 1

$$\ell(\mathbf{x}\,|\,\mathcal{M}) = \sum_{i_1,i_2} \log \left(\sum_{c\in S_{i_1,i_2}} \pi_c g(\mathbf{x}_{i_1,i_2}\,|\,c) \right) \enspace . \tag{11}$$

Combining Eq. 11 and the inequality in Eq. 10, the conclusion follows.

□

The density defined in Eq. 1 is a convex combination of the densities in its sum. Therefore,

$$f(x|S) \leq \max_{c\in S} g(x|c) = b_{x,S} \enspace . \tag{12}$$

This bound is a function of x and of S. The bound for the log-likelihood $\ell = log\ L$ as

$$\ell(\mathcal{M}) = \sum_{i_1,i_2} \log f(x_{i_1,i_2}\,|\,S_{i_1,i_2}) \leq \sum_{i_1,i_2} log b_{x_{i_1,i_2},S} \enspace , \tag{13}$$

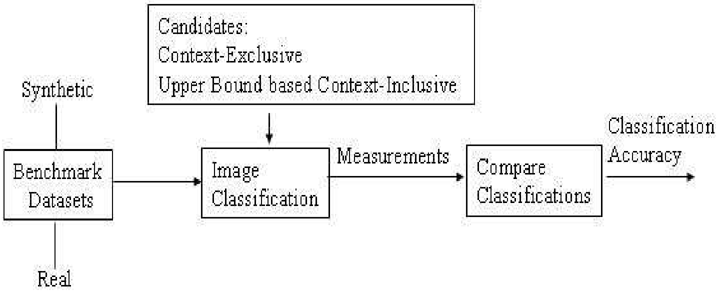


Figure 6: Experimental Design

since the logarithm function in monotonic increasing which is a model-specific bound. A bound over all models is

$$\max_{\mathcal{M}} \ell(\mathcal{M}) \leq \sum_{i_1,i_2} logb_{x_{i_1,i_2}} \quad . \tag{14}$$

Both of these bounds are clearly finite if the class-specific densities $g(x|c)$ are finite for all x.

For example, suppose we want to determine the upper bound for $Forest$ in Table 2 for the given likelihood values. First, we would find the maximum value between the two specific classes ($conifer$ and $hardwood$) for each pixel (Table 6). Then, we would take the sum of each value in Table 6 which will give the upper bound for $Forest$. Thus, the upper bound for $Forest$ will have a likelihood of 3.4 or a log-likelihood of 1.2238. Recall that in Section 2.2, the likelihood for $Forest$ after executing EM is 1.17162 or the log-likelihood of 0.1584 which shows that the upper bound found in this example is correct.

0.8	0.8
0.9	0.9

Table 6: Maximum value between the specific classes $conifer$ and $hardwood$.

4.2 Proof of Correctness of Upper Bound based Context Inclusive Approach

Theorem 1 *The Upper Bound based Context Inclusive approach (UBCI) is correct such that each pixel in the image is classified with the maximum likelihood or best candidate class from the user-defined concept hierarchy.*

Proof Based on Algorithm 3, an upper bound is assigned to each non-specific class candidate. The candidate with the highest upper bound is chosen first and the converging value based on EM is assigned as the initial best candidate (Lines 2-4 of Algorithm 3). If the upper bound for every other candidate is less than the current best candidate's likelihood, then the candidate will not ever have a higher likelihood value and can then be pruned from the list (Lines 5-8 of Algorithm 3). All other candidates having a higher upper bound than the current best candidate may have a higher actual likelihood value (from EM), which is then assigned as the best candidate (Lines 10-14 of Algorithm 3). Thus, the best correct candidate will be retrieved from the candidate list. Also, the output of the UBCI and the Context Exclusive approaches are identical. □

5 Experimental Evaluation

The goal of our experiments was to compare the Upper Bound based Context-Inclusive approach with the Context-Exclusive approach in terms of computation efficiency. Computation efficiency is measured at a physical (computation time) and at a logical (EM Iterations) levels. Figure 6 provides a schematic representation of our experimental design. All experiments were performed on two different datasets: (1) A synthetic input image having the size of , 7 total class, and 3 *general* classes (Figure 2); and (2) A real dataset of Plymouth County,

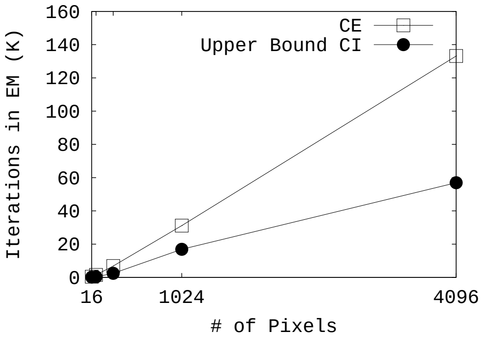


Figure 7: Iterations in EM for the Synthetic Dataset

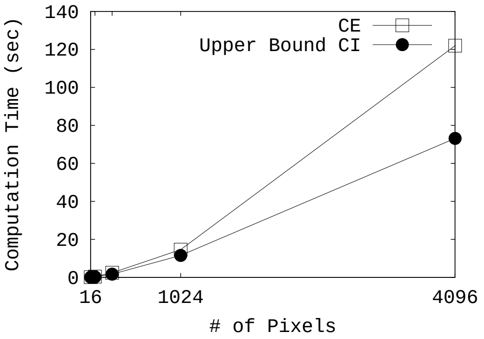


Figure 8: Computation Time for the Synthetic Dataset

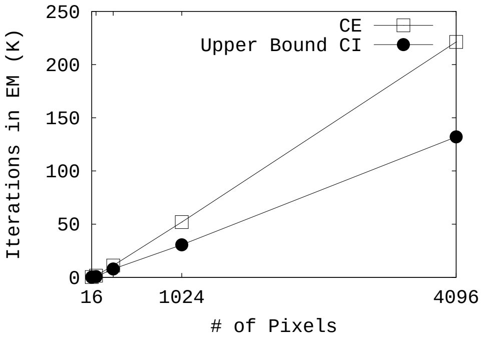


Figure 9: Iterations in EM for the Real Dataset

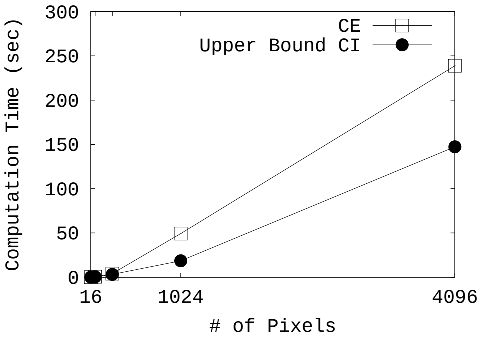


Figure 10: Computation Time for the Real Dataset

Massachusetts having 12 total classes, and 4 *general* classes. Both images have the size of 64 x 64 where multiple image sizes were analyzed (i.e., 16, 64, 256, 1024, 4096 pixels). Outputs were obtained for varying the number of pixels that was analyzed by each proposed approach. All experiments were performed on an UltraSparc III 1.1 GHz processor with 1GB of RAM.

5.1 EM Iterations

As discussed in Section 1, evaluation of the quality measure takes up most of the time. Hence the number of iterations in EM may be used to evaluate the computation of the Upper bound based context-inclusive and context-exclusive algorithms. Experimental results of both approaches at multiple spatial scales on the synthetic and real datasets are shown in Figures 7 and 9, respectively. As the number of pixels increase, the number of iterations increases as well because of the larger size of regions. Note that the real dataset has more *general* classes than the synthetic, and thus is more computationally expensive. Experimental results show that the number of iterations is less for the Upper Bound based context-inclusive approach because several non-specific classes may be pruned due to its upper bound being less than the current best class. Compared to the context-exclusive approach, at the largest number of pixels, the number of iterations with the upper bound context-inclusive approach reduces by 57.2% for the synthetic and by 40.4% for the real dataset.

5.2 Computation Time

Figures 8 and 10 provides the execution time taken for the synthetic and real datasets, respectively. The computation time was measured using MATLAB 7 and using the functions “clock” and “etimeof” to capture the elapsed time for both the Context-Exclusive and the Upper Bound based Context-Inclusive approaches. Since the number of iterations taken by the upper bound based context-inclusive approach is less than that for the context-exclusive approach, the execution time for the context-inclusive approach will also be less. Since the number of models that are evaluated by the upper bound based approach will likely be less than the context-exclusive approach. As the number of pixels in an image increases the number of regions that need classification also increases. This results in the need of more non-specific classes that need to be analyzed resulting in the significant increase in the context-exclusive approach, whereas in the upper bound based context-inclusive approach, several models are pruned resulting in a much lower execution time. At the largest image size, the execution time for the upper bound based context-inclusive approach, as compared to the context-exclusive approach, is reduced by 40% and 38.3% for synthetic and real datasets respectively.

6 Discussion and Future Work

We presented an upper bound based context-inclusive approach (UBCI) that utilizes an upper bound for each candidate model to allow for pruning to obtain efficient and correct results. Also, we proved that an upper bound exists for each candidate model and experimentally showed our dominance over the context exclusive approach. Our approach is based on set context where tuples are evaluated together to obtain the optimal candidate model. Other types of context may be explored such as spatial context i.e., the correlation of a variable with space [13], may be used. Classification of spatial data based on an extended regression model called the Spatial Auto-regression model (SAR) is provided in [14].

As discussed, most of the execution time is spent in calculating the quality measure using EM. EM is used to find the best candidate model; more specifically, EM is used to find the best Gaussian mixture model in the case of land-use classification. We plan to explore faster implementations of EM such as [3, 4, 11]. Larger datasets may be used to extend the experimental evaluation along with alternative measures such as I/O costs. Also, further optimizations of the UBCI may be explored such as ordering the non-specific classes based on their upper bounds.

Acknowledgments

This work was a result of research supported in part by grants from NSF SEI ISS-0431141, NSF BCS 0079077 and 0318209, Office of Naval Research award N00014-99-1-0219, and IGERT. We thank the group members of the Spatial Database and Data Mining group at the University of Minnesota. We are grateful to Kim Koffolt for improving the readability of this paper. Finally, we thank the reviewers of our paper at the Spatial and Spatial Temporal Data Mining workshop for providing valuable suggestions.

References

[1] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the em algorithm. volume Series B, 39(1), pages 1–38. Journal of the Royal Statistical Society, 1977.

[2] V. Gandhi, J. M. Kang, S. Shekhar, J. Ju, E. D. Kolaczyk, and S. Gopal. Context-inclusive approach to speed-up function evaluation for statistical queries : An extended abstract. In *ICDM Workshop on Spatial and Spatio-temporal Data Mining (http://www.cs.hku.hk/ nikos/sstdm06/).*, pages 371–376, 2006.

[3] H.-S. Huang, B.-H. Yang, and C.-N. Hsu. Triple jump acceleration for the em algorithm. In *ICDM '05: Proceedings of the Fifth IEEE International Conference on Data Mining*, pages 649–652, Washington, DC, USA, 2005. IEEE Computer Society.

[4] A. Ihler. *Inference in Sensor Networks: Graphical Models and Particle Methods*, chapter Ph.D. Thesis, pages 26–32. Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 2005.

[5] J. R. Irons, B. L. Markham, R. Nelson, D. L. Toll, and D. Williams. The effects of spatial resolution on the classification of Thematic Mapper data. In *International Journal of Remote Sensing*, volume 6, pages 1385–1403. Taylor and Francis, 1985.

[6] J. Ju. *Scale effects in remote sensing: Sub-pixel and supra-pixel land cover characterization.* PhD thesis, Geography and Environment and Center for Remote Sensing, Boston University, 2005.

[7] J. Ju, S. Gopal, and E. D. Kolaczyk. On the choice of spatial and categorical scale in remote sensing land cover classification. In *Remote Sensing of Environment*, volume 96, page 6277. Elsevier Science, 2005.

[8] J. Ju, E. D. Kolaczyk, and S. Gopal. Gaussian Mixture Discriminant Analysis and Sub-Pixel Land Cover Classification in Remote Sensing. In *Computing Science and Statistics*, volume 34. Elsevier Science, 2002.

[9] E. D. Kolaczyk, J. Ju, and S. Gopal. Multiscale, Multigranular Statistical Image Segmentation. In *Journal of the American Statistical Association*, volume 100, pages 1358–1369, 2005.

[10] B. L. Markham and J. R. G. Townshend. Land cover classification accuracy as a function of sensor spatial resolution. In *Proceedings of the Fifteenth International Symposium on Remote Sensing of Environment*, pages 1075–1090. Elsevier Science, 1981.

[11] G. McLachlan. On aitken's method and other approaches for accelerating convergence of the em algorithm. In *Proceedings of the A.C. Aitken Centenary Conference*, pages 201–209. University of Otago Press, 1995.

[12] V. S. Raptis, R. A. Vaughan, and G. G. Wright. The effects of scaling on land cover classification from satellite data. In *Computers & Geosciences*, volume 29, pages 705–714. Elsevier Science, 2003.

[13] S. Shekhar and S. Chawla. *Spatial Databases: A Tour*. Prentice Hall, 2003.

[14] S. Shekhar, P. Schrater, R. Vatsavai, W. Wu, and S. Chawla. Spatial Contextual Classification and Prediction Models for Mining Geospatial Data. In *IEEE Transaction on Multimedia*, 2002.

[15] Wikipedia. Convex combination, October 1, 2007. http://en.wikipedia.org/wiki/Convex_combination.

[16] A. Willsky. Multiresolution Markov models for signal and image processing. In *Proceedings of the IEEE 90*, volume 8, pages 1396–1458, 2002.