DESIGN PRINCIPLES FOR INTERACTIVE IMAGE INFORMATION MINING SYSTEMS

G.K. Ouzounis

1 DigitalGlobe Inc., 1601 Dry Creek Drive, Suite 260, Longmont, CO 80503, USA.
georgios.ouzounis@digitalglobe.com

Modern image information mining systems operate on vast amounts of data making the efficient organization and management of the image information content to be of crucial importance. Addressing this challenge, tree-based image representation structures were introduced to allow for the precise definition of image components, to shape up component hierarchies and to provide efficient means for component attribution. Trees are used as interfaces between the image space and some control module often hosting a feature space. Making use of the advanced search and retrieve mechanisms of most tree algorithms a new system architecture and an operation protocol for interactive image information mining are proposed. They are demonstrated using the Alpha-Tree structure on a built-up mining task from very high resolution panchromatic satellite imagery.

Introduction

Image information mining (IIM) systems are found in a wide range of modern applications such as online search engines, intelligence gathering platforms, diagnostic tools in medicine, etc. IIM systems facilitate query mechanisms over an input data space. A query is a supervised procedure in which a user or a system requests the detection of image features that satisfy certain criteria such as semantic labels, thresholds on object attributes or enumerated resemblance to one or more target prototypes from a set of examples [1-3].

The architecture of IIM systems varies depending on the application, however there exist three key requirements common in each design; the definition of a structured search space, the definition of an object attribution or semantic characterization schema, and the definition of a search and retrieve mechanism [2]. All three are crucial in the design of a functional system and each has its own share on the overall performance, especially as the volume and complexity of the input data increases. It is often the case that each requirement drives the design of a separate subsystem. This increases the overall system complexity at the cost of reduced performance. This talk gives an insight into design principles from a single system perspective and introduces a new IIM paradigm based on hierarchical image representation structures.

The Anatomy of an IIM System

The definition of a structured search space requires the organization of the image information content into meaningful entities or components. Shaping up such components for the entirety of the image space compresses the information content and improves the performance of the query mechanism. The set of all elementary components, i.e. those in the initial allocation, is called a base layer. The search space may coincide with the base layer or consist of multiple layers, capturing in this way hierarchical relations between components according to some pre-defined nesting order. Fig.1. shows a test pattern on the white frame and the corresponding hierarchical image representation structure, an Alpha-Tree [4] that coincides with the search space. Components are shown as white dots interlinked by red wires.

The second requirement concerns the way components are represented. Components are attributed during or after the organization of the image information content.
Attributes are often geometric, radiometric or other statistically derived measures that describe each component in question. The set of attributes associated with each component is used by the search and retrieve mechanism when comparing an image feature against a target prototype. Thus the better the attribution is, the better the output of the query will be.

The query mechanism is often a set of logical predicates applied on each examined component. If the component satisfies the query criteria it remains intact otherwise it is removed in its entirety. Both actions are executed by rearranging spatial and/or hierarchical relations between the components involved.

**The Worst Case Scenario**

Given a complete system we now touch upon the scenario in which one or more of these requirements is not met or not implemented properly.

The lack of structure or poorly defined structure of the information content and thus of the search space means that components must be formed during the query run-time. This introduces major bottlenecks and makes queries time-consuming, resource-demanding and prone to error as they become sensitive to region-access-order. Moreover, repeated queries for fine tuning the output make the search process prohibitive.

The attribution is mostly responsible for the quality of the query results. A target prototype that is poorly described in terms of attributes may be the cause of severe noise in the output. By contrast, over constrained characterization will often fail to generalize appropriately, limiting the target class variability and thus requiring more same-class prototypes.

The search and retrieve mechanism is interrelated to the search space organization thus any poor design has an impact primarily on the system performance.

**Hierarchical Image Representation**

Complying with all three requirements in the design of a complete system, we end up with what is usually referred to in image analysis as a hierarchical image representation data structure or simply a dendrogram. Examples of known dendrograms are the Max-Tree [5], the Alpha-Tree [4], the Binary Partition Tree [6], and others. The key difference between different types of trees is the set of rules defining the base layer and the evolution parameter shaping up the respective hierarchy.

The Alpha-Tree [4] was demonstrated as an IIM system in [2] giving evidence of being a powerful tool. In brief, an Alpha-Tree organizes the image space into a stack of nested partitions, each defined at a given dissimilarity threshold $\alpha$. Each partition cell is an $\alpha$-component which is essentially an image region consisting of pixels for which, any two that are adjacent have a difference of less than or equal to $\alpha$ according to some pre-defined dissimilarity metric. The Alpha-Tree maps only the first instance of an $\alpha$-component from the stack of partitions, eliminating this way redundancies in the representation. The leaves of the tree correspond to components consisting of elements that are fully similar to each other with respect to the dissimilarity metric. The set of leaves defines the base layer in the IIM search space. The root of the tree corresponds to the single $\alpha$-connected component that coincides with the image definition domain.

Each node of the tree maps to a unique $\alpha$-component from the stack of partitions and points to its parent; the first superset of the given $\alpha$-component at $\alpha'$ strictly greater than $\alpha$. 

---

Fig. 1. A 3-D view of the Alpha-Tree computed from the 10-level test pattern shown in the white frame. The Alpha-Tree defines the search space for the test pattern.
Moreover, each node contains a pool of auxiliary data that are updated incrementally during the tree construction phase. Based on this data, component attributes can be computed in real time and during every query. Nodes are indexed making search and retrieval mechanisms fast and efficient.

A Protocol for Interactive Image Information Mining

Assume an Alpha-Tree defining an IIM system architecture. Image operators are computed on an Alpha-Tree with a single pass through the structure. Node attributes are computed upon visiting a node from the pool of auxiliary data it contains.

This functionality allows for real-time multi-dimensional feature space computation.

An example for the input image of Fig.3(a) is shown in Fig.3(b). This is an α-connected pattern spectrum [7] with the origin at its top left corner. The x-axis corresponds to a logarithmic mapping of component size and the y-axis to a square-root mapping of the component compactness measure. The energy of each bin, shown in heat-map color code, corresponds to a normalized contribution of the image information content; i.e. the pixel count of all components (represented by nodes) satisfying the bin extreme conditions multiplied by each node’s lifetime.

If target prototypes are to drive a query, they must be first defined. Prototypes can be imported from an external data-base or collected directly from the image space. In both cases a binary mask is used to mark all maximal components that are fully included in the mask’s foreground extent. For each external prototype an individual tree is constructed from which component attributes are computed. For prototypes collected from the input image space, a visit to each node
associated with a detected maximal component allows for direct attribute computation.
The attributes of all components identified as target prototypes are then projected onto the
feature space indicating which regions are of interest. Fig.3(c) shows this for all targets
within the green masks in Fig.3(a). The dominant cluster at the left of the plot that
stretches along the y-axis shows the contribution of side components to our targets,
which are small in size, of varying compactness and captured under the mask.
Ignoring them and focusing at the remaining points, a second smaller cluster is found at the
bottom of the plot and in the middle of the x-axis. This cluster marks a wider region of
interest according to the heat map in Fig.3(b), suggesting the presence of a persistent pattern.
Using the switchboard interface [2] a pilot query is launched by activating the switches
associated with the bins in this area of interest. Fig.4(b) shows the activated switches in green.
A pass through the tree collects and projects the attributes of each node onto the feature
space. If the switch of the corresponding bin is set on the associated component is retained in
the query output, otherwise it is ignored. This generates a segmentation and an example for
the image in Fig.3.(a) is shown in Fig.4(a).

Discussion

Revising the proposed architecture of an IIM system we observe the following facts: (i) The
tree construction is fast and offers a compact and explicit representation of the image
information content. (ii) The feature space can be computed rapidly [8] and further be used
for training a classifier. (iii) Classifiers can be computed directly on the tree structure [3]. (iv)
The search and retrieve mechanism comes with each tree algorithm and is very efficient.

The most important feature of this architecture is the ability to launch queries on adjustable
number of target prototypes without the need of system recalibration. Adding or removing
prototypes translates into projecting their attributes onto the feature space and since the
query is controlled via the switchboard the system performance remains unaffected.

Conclusions

The adaptation of a tree data structure for implementing an IIM system complies with all
the design requirements discussed. The system benefits from all the additional tree
functionalities and supports rapid queries with varying numbers of target prototypes.

References

1. L. Gueguen, G. K. Ouzounis. Tree-based representations for fast information mining from
2. G. K. Ouzounis. V. Syrris. L. Gueguen. P. Soille. The switchboard platform for interactive image
3. L. Gueguen, G. K. Ouzounis. Hierarchical data representation structures for interactive image
5. P. Salembier. A. Oliveras. L. Garrido. Anti-extensive connected operators for image and
6. P. Salembier. L. Garrido. Binary partition tree as an efficient representation for image processing,
7. G.K. Ouzounis, P. Soille. Pattern spectra from partition pyramids and hierarchies. // Springer,
pattern spectra for very large image information mining. // ESA-EUSC-JRC Proc. 8th Int. Conf.

About the author

Dr. Georgios K. Ouzounis specializes in mathematical modeling and algorithm development/optimization for
geospatial image analysis. From September 2009 to September 2012 he worked as a researcher at the Global
Security and Crisis Management Unit of the Joint Research Centre of the European Commission in Ispra,
Italy. He is with DigitalGlobe since October 2012.

Georgios K. Ouzounis has invented three major algorithms: the dual-input Max–Tree algorithm, the
Area Zone Decomposition for computing Differential Area Profiles (DAPs), and the Alpha-Tree. He was the
architect of all computational engines related to the DAP vector fields that were used by the JRC Image
Query (IQ) system to populate the first VHR Global Human Settlement Layer (GHSL).