# A Framework for the Representation of Geospatial Image Processing Operations


Vijay Venkataraman, Peggy Agouris
Department of Spatial Information Science & Engineering, University of Maine
348 Boardman Hall, Orono, ME 04469
{vijay, peggy}@spatial.maine.edu



**Abstract**

*Research advances in geospatial automated image analysis tools and feature extraction algorithms have matured in recent times to levels of practical applicability. The consolidation of such tools and algorithms would result in enhanced image analysis capabilities. This has motivated research in developing formalisms for representation of process information that can assist in integrating tools and data sets, and allow for effective dissemination of research results across a raster-rich geospatial distributed information system. We propose a framework for a formal representation of image processing operations and algorithms as process metadata with generalized adaptability across all raster data sets. The elements of such a formal representation and their significance in the domain of distributed information systems are presented.*


## 1. Introduction

Geospatial raster data sets present a source of rich information with specific characteristics that differentiate them from any other generalized raster data set. They are feature rich with specific *shape and configuration enumerations* that allow for development of technologies that are restricted to their domain. Moreover, geospatial imagery is rarely captured and merely stored. It is typically subject to intense processing, in a manual, semi-automated or automated fashion to extract relevant geospatial information.

Development of automated image analysis and feature extraction tools has been the focus of substantial research within the geospatial raster research community (Doucette, 2001), producing algorithms that are reusable across data sets. These algorithms are of varied nature. Some focus on feature extraction, while others focus on querying for content. With anticipated explosion of high quality satellite and aerial imagery, the dependency on automated and semi-automated tools has significantly increased as processing and analysis requirements increase beyond manual thresholds.

In addition, sharing geospatial raster data sets through traditional peer-to-peer (P2P) networks is becoming prevalent. Several concepts have been suggested for cataloguing and querying image databases. Prior efforts, like MetaSEEK (Beigi, 1998), rely heavily on image metadata as the basis of cataloging and performing queries. Systems like Blobworld (Carson, 1999) attempt to locate coherent image regions and approximate them to objects and match them to a user's query. Attempts at describing complete systems have focused more on optimizing cataloging and querying techniques rather than system functionality (Petrakis, 1993).

A formal representation of image processing operations as generalized process metadata would thus enable a fast and accurate mechanism of introducing research and development into a larger community and would help transform a static data cataloguer into a dynamic system that is capable of delivering answers tailored to end user queries. The formal representation takes into account that process metadata can be an amalgamation of multiple contributions. This enables the process metadata to be extended to different levels of derivation for specific applications and at the same time can accommodate different types of data sets including, but not limited to, image data, motion video and data from geospatial sensor networks. In the process we explore the framework only from the point of functionality and not from those of performance or technical implementation standards.

The rest of the paper is organized as follows: Section 2 discusses Metadata, Section 3 presents the framework, Section 4 discusses a Raster-rich Geospatial Distributed Information System (RGDIS) with a conclusion in Section 5.

## 2. Metadata

Metadata, in the traditional sense of the word, is data about data in terms of meaning, content, organization, or purpose of data (Liu, 1997). Metadata can be broadly classified into *logical* and *machine-interpretability* representations. The former defeats the scope of having metadata as a means to achieve automation. Hence machine-interpretability has taken precedence over logical representation and the metadata is primarily used as a mechanism for communication between disparate components or systems. A middle of the ground approach adopts metadata at both the logical and machine levels, with explicit relations between the two for easy inter-translation.

Metadata is also used for representation at different levels of *granularity* and *perspectives* (Vassiliadis, 1998). When granularity is global, metadata is most representative of the conceptual or logical level (*logical perspective*) and at atomic granularity, is more machine interpretable (*physical perspective*). This approach has lead to the various models of defining and structuring metadata for different application domains and query models. In addition to this, metadata schemes have also been further enhanced by incorporating semantics and ontology (Bornhövd, 1999) using ontology as a common interpretation basis for simple semantic objects.

Image processing operations have been represented in many commercial desktop or workstation applications as *program scripts*. However, they have hardly been designed in the context of distributed information systems, and generally use proprietary interfaces and representation schemes that will work only within the bounds of the concerned software environment (e.g., MATLAB Image Processing Toolbox and Scripts). We redress this shortcoming by proposing to extend the scope of metadata to define processes, and *treat process metadata independent of any specific object*. This allows us to generalize process metadata and introduce a representation framework for processes that would be valid across entire raster data sets.

## 3. Elements of Formal Representation

Any formal representation of image processing operations should address issues related to distributed information systems as their primary application domain. Fig 1. shows the elements of the formal representation which allows for different types of metadata coexisting in the same system. *Images* in a database system form the raw data and are associated with *descriptive metadata*. The dynamic *process metadata*, independent of any particular image, can be used to derive processed images on request. Process metadata can be nested or called from other process metadata and typically contain a sequence of steps of an operation. To support this, a basic syntax and toolbox containing standard functions need to be designed on which more complex algorithms can be built.

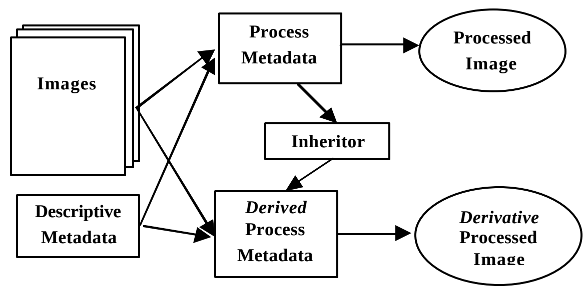


**Fig 1. Elements of Formal Representation.**

A process metadata can be approximated syntactically as follows:

*<header>*
*<input specifications>*
*<constraint specifications>*
*<operation code: tool identifier>*
*<output specifications>*
*<flag>*

The *header* identifies the type of metadata and also contains a unique identifier within the system, and can contain information on physical location and security associations. *Input* *specifications* are used to gather necessary inputs, including the object to be processed. *Constraint specifications* immediately follow the input specifications which check if the current process metadata can be applied on the object. *Operation code* contains the actual processing code. A *tool identifier* is contained within the code specifying any built-in or externally referred tools *or process metadata* to perform an operation. After processing, the object state and associated descriptive metadata are updated to reflect changes. Every operation sequence may individually recheck the updated metadata for constraint satisfaction. The *output specification* returns the object after successful operation. A *flag* is used to signal the result of an operation and can also be used to return a confidence level of output based on any custom metrics used to measure the quality of input and output.

A classic example of a process metadata is given below:

```
<metadata type=”process” id=”{unique id }”>
        <input param=”{obj}, {par1 }”>
        </input>
        <constraint param=”{obj}”>
                ……
        </constraint>
        <constraint param=”{par1 }”>
                ……
        </constraint>
        <process>
                <opr toolid=”{unique id 1}” param=”{obj},
{par1 } output={out1 }” />
        </process>
        <output result=”{out1 }” />
        <result toolid=”{unique id} param=”{obj}”
default=”1” />
</metadata>
```

The framework of a process metadata system should have the following elements:

- *Header*: Every metadata contains a header that describes in detail the type of metadata. Descriptive and process metadata are distinguished through their headers
- *Standard Process Metadata Interface*: A standard interface for the process metadata allows an object reference can be passed regardless of the type. Likewise, it should also contain a standard result interface that can return a basic response to a request.
- *Process Metadata Constraint Definition*: Every process metadata has built in a constraint definition which decides if the operations are possible on an input object. If the operations are possible then the sequence of operations are carried out and the result returned. If the input object does not satisfy the constraints, then a negative response is returned.
- *Object States*: The purpose of an object state is to uniquely identify the object within the system and track the evolution of the object with progression through different process metadata operations. A history of the object states can also be maintained to enable rollback.
- *Code and Platform independency*: Process metadata can refer to external binary toolsets that have been specifically developed for use on raster data sets. This has two practical advantages: (1) It can accommodate both open and closed source systems, and (2) it can bypass the limitations of a metadata system by embedding intensive operations in toolboxes wrapped with the process metadata classes
- *Inheritance*: A previously defined process metadata to be enhanced further through derivations for further specialized operations. In this case, the derived process metadata would reference the parent process metadata and add further operations to the result of the parent process metadata. Inheritance can also be inherited from multiple parent metadata.
- *Representation standards*: Standards (e.g. XML) used to represent the metadata should comply with platform independence and must be internet enabled.

## 4. A Raster-rich Geospatial Distributed Information System (RGDIS)

Digital collections of geospatial data sets and their public dissemination have been a popular theme among researchers in the last decade. The information to be shared is indexed through metadata and queries are run on them to extract relevant data sets for sharing.

A classic example of the benefits of introducing our formal representation framework in a RGDIS system is shown in Fig. 2. In this case, *User1* (*data provider*) contributes data and descriptive metadata to the RGDIS system. *User2* (*tool provider*) provides a process metadata to do a particular image processing operation that can be done on the data set provided by *User1*. Another user, *User3* further enhances the tool provided by *User2* to give a derived process metadata output. The benefits of a multi-user contribution to a single distributed information system are immediately apparent. Not only is data being shared, but the scope of the system has been enhanced further by the sharing of analytical capabilities.

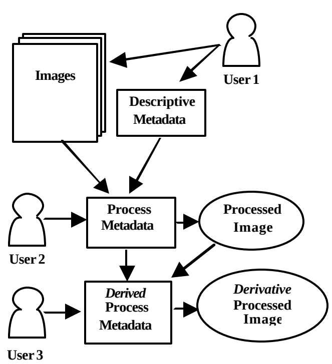


**Fig 2. Advantage of introducing formal process metadata into RGDIS.**

## 5. Conclusion

The explosion of available geospatial imagery likely from multitude of sources, including the IKONOS panchromatic images and the polarimetric radar images of the RADARSAT and JERS satellites, has resulted in a need for sharing data and analysis tools through a centrally managed system. The research efforts of the professional geospatial image processing community remain largely scattered and disorganized.

The framework proposed by this paper attempts to develop a multi-tier distributed image database system. Several technical issues need to be addressed, before an actual implementation can be envisioned. The proposed framework compromises performance to achieve extended functionality and necessitates optimization techniques. With exponential increase in computing power, there is strong motivation to extend traditional computational systems into areas offering rich feature sets and functionality.

## Acknowledgements and Copyright


This work is supported by the National Science Foundation through grants no. EIA-9983445 and IIS-0121269.


## References


[1] P. Doucette, P. Agouris, M. Musavi, A. Stefanidis, *Self-organised clustering for road extraction in classified imagery*, ISPRS Journal of Photogrammetry and Remote Sensing, Elsevier,Vol.55, No. 5-6, pp. 347-358, Mar 2001

[2] M. Beigi, A. B. Benitez, S. Chang, *MetaSEEk: A Content-Based Meta-Search Engine for Images*, IST/SPIE 1998, Vol. 3312

[3] C. Carson, M. Thomas, S. Belongie, J. Hellerstein, J. Malik (1999), *Blobworld: A system for region-based image indexing and retrieval*, 3rd Intl. Conf. on Visual Information Systems

[4] E. Petrakis (1993), *Image Representation, Indexing and Retrieval Based on Spatial Relationships and Properties of Objects*, Ph.D, Dept. of Computer Science, University of Crete.

[5] L. Liu and C. Pu. (1997), *A Metadata-based Approach to Improving Query Responsiveness*, in Proceedings of the 2nd IEEE Metadata Conference. Sept, 1997

[6] P. Vassiliadis, Y. Stavrakas (1998), *Different Perspectives of Metadata for Web-based Information Systems*, 11th ERCIM Database Research Group Workshop on Metadata for Web Databases, Sankt Augustin, Germany 25-26 May 1998

[7] C. Bornhövd (1999), *Semantic Metadata for the Integration of Web-Based Data for Electronic Commerce*. WECWIS 1999: 137-145