LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

UCRL-CONF-204846

Computer-Aided Content-Based Cueing of Remotely Sensed Images with the Image Content Engine (ICE)

G. F. Weinert, J. M. Brase, D. W. Paglieroni

June 23, 2004

International Geoscience and Remote Sensing Symposium (IGARSS)
Anchorage, AK, United States
September 20, 2004 through September 24, 2004

Disclaimer

Computer-Aided Content-Based Cueing of Remotely Sensed Images with the Image Content Engine (ICE)*

George F. Weinert, James M. Brase and David W. Paglieroni
Lawrence Livermore National Laboratory
P.O. Box 808, L-290, Livermore, CA 94550

*This work was performed under the auspices of the U.S. Department of Energy by the University of California, Lawrence Livermore National Laboratory, under contract No.W-7405-Eng-48

Abstract - Human analysts are often unable to meet time constraints on analysis and interpretation of large volumes of remotely sensed imagery. To address this problem, the Image Content Engine (ICE) system currently under development is organized into an off-line component for automated extraction of image features followed by user-interactive components for content detection and content-based query processing. The extracted features are vectors that represent attributes of three entities, namely image tiles, image regions and shapes, or suspected matches to models of objects. ICE allows users to interactively specify decision thresholds so that content (consisting of entities whose features satisfy decision criteria) can be detected. ICE presents detected content to users as a prioritized series of thumbnail images. Users can either accept the detection results or specify a new set of decision thresholds. Once accepted, ICE stores the detected content in database tables and semantic graphs. Users can interactively query the tables and graphs for locations at which prescribed relationships between detected content exist. New queries can be submitted repeatedly until a satisfactory series of prioritized thumbnail image cues is produced. Examples are provided to demonstrate how ICE can be used to assist users in quickly finding prescribed collections of entities (both natural and man-made) in a set of large USGS aerial photos retrieved from TerraserverUSA.

I. OVERVIEW OF ICE

The Image Content Engine (ICE) is being developed at the Lawrence Livermore National Laboratory as a system for assisting human analysts in timely extraction of content and knowledge from large volumes of remotely sensed imagery. It supports automatic extraction of features from images, followed by user-interactive detection of entities of interest stored in tables or semantic graphs, followed by user-interactive table/graph query processing for generating cues to regions-of-interest containing entities with prescribed relationships to one another (see Fig.1).

Like other systems, such as PicHunter [1] or SOM-AIR [2], ICE relies on an offline, compute intensive feature extraction process to provide source material for its databases. However, unlike other systems, ICE treats object detection as a separate user-interactive process that follows automated feature extraction. By inserting the output of user-interactive detection into database or graph structures, the quality of data stored improves, along with the likelihood of successful subsequent queries. Unlike PicHunter, ICE does not currently include the ability to provide relevance feedback during queries, although it is our goal to include this at a later date.

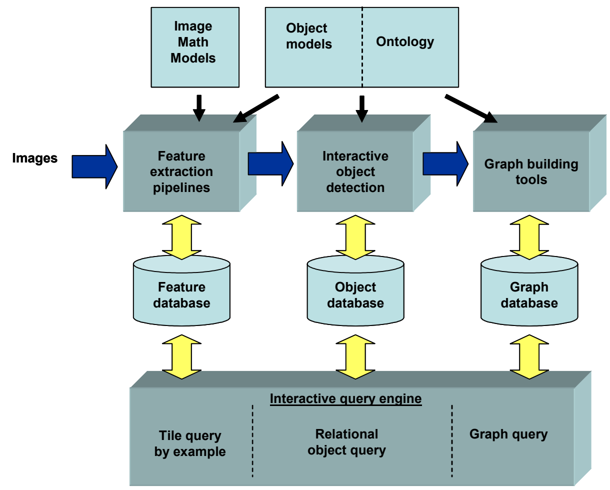

Fig.1 ICE overview.

ICE deals with three types of entities: tiles, regions, and objects. ICE automatically extracts user-specified features for each type of entity off-line. Feature extraction is the most computationally expensive portion of the analysis process, and may require parallel processing in order to achieve satisfactory pixel throughput rates.

Tiles are square evenly spaced adjacent or overlapping image blocks (often small) from which feature vectors are extracted. Tile feature vectors form the basis for classifying image tiles by type (e.g., forested).

Regions are extracted by using an image segmentation algorithm to group similar pixels together. Region feature vectors capture generic spatial and spectral properties of regions. Region feature vectors may also contain degrees of similarity (from 0 for poor similarity to 1 for perfect similarity) to specific shapes, such as ellipses or specific types of polygons (selected from a shape registry) so that regions may be later classified by shape type. Regions are matched to shapes of specified type and arbitrary position-size-orientation using either the method of moments [3] or boundary matching [4].

Objects consist of one or more 3D components that can be spatially separated and are typically man-made (e.g., trucks, facilities that contain a particular arrangement of specific types of buildings, etc.). Using a novel phase sensitive matcher [5], projections of 3D object model edges onto the image (i.e., 2D object signatures) are matched to the image over all combinations of object position and orientation to produce one match similarity (from 0 to 1) at each position. State vectors, consisting of match similarity, position and orientation, are stored as feature vectors for each local maximum in similarity that is not in conflict with other local maxima.

ICE supports user-interactive detection based on features and match similarities that ICE previously obtained from images automatically in the feature extraction phase. This provides human analysts the opportunity to regulate the content that gets stored in database tables and semantic graphs for later query and retrieval. The analyst is allowed to interactively set thresholds on similarities between extracted entities and models of target entities. ICE retrieves thumbnail images from a designated set of search images that contain detected entities (content), and then sorts them by figure-of-merit. The analyst can then decide which thumbnails correspond to valid detections so that only they get stored in database tables and semantic graphs. Currently, the analyst can only specify a point in the thumbnail list above which all thumbnails are considered valid detections.

In the future, ICE will support user-interactive query of content databases and semantic graphs as a means of analyst cueing. However, an evolving query capability currently exists. Analysts can specify how many detected entities of various types must be present in regions-of-interest, and can even provide some information about how the detected entities must relate to one another. The semantic graphs capture information pertaining to local relationships between extracted content. These relationships include “parallel to”, “in series with”, “connected to”, “close to”, etc. The types of relationships captured are part of a growing list, and together with registries of admissible shapes and objects, characterize the ICE ontology.

II. FEATURE EXTRACTION AND MATCHING

The ICE feature extractor is a user-configurable application. The input is a set of images to be searched, and the output is a set of files of extracted features and matches. The user can specify which tile features to extract, and which shapes and objects to match. ICE converts this specification into a set of software pipelines. One pipeline handles all tiles, another pipeline handles all regions/shapes, and one pipeline is assigned to each object type (see Fig.2). Each pipeline corresponds to one pass through the images to be searched. For each pipeline, images are broken down into square *blocks*. Image block width is user-specified and determined by image spatial resolution and object size. Image block overlap is either user-specified (for regions/shapes), or computed automatically (for tiles and objects).

The regions/shapes pipeline preprocesses the image block (this involves brightness-contrast enhancement, de-speckling [6] and/or quantization), segments the block (using a coarseness regulation technique [7] that greatly improves segmentation quality [8]) and matches each region to one or more designated shapes stored in a shape registry. The tiles pipeline divides image blocks into tiles of designated size/overlap and extracts designated features from each tile. The object pipelines use an FFT-based phase sensitive detector [5] to efficiently match projections of 3D models of object edges to the image at each position and orientation. The user selects which objects to match from an object registry. The output of phase sensitive matching is a surface of match similarities within which unambiguous local maxima are found. A local maximum is said to be unambiguous if its object match does not overlap the object match associated with any other local maximum of greater or equal similarity value. State vectors (consisting of match similarity, position and orientation) are assigned to each unambiguous local maximum. ICE automatically computes the amount of block overlap needed so that objects will never be missed due to the fact that there is no block that they lie completely within. Since the minimum required overlap increases with the extent of object projections in image space, and different objects often have different extents, each object typically requires a different block overlap resulting in different sets of blocks, and different sets require different pipelines. Note that for each object and image block, there is one set of model projections. The object is translated so that its centroid matches the geo-coordinates of the block center. The object is then rotated over a full 360° set of rotations about the z axis, and each rotation is separately projected into the image block.

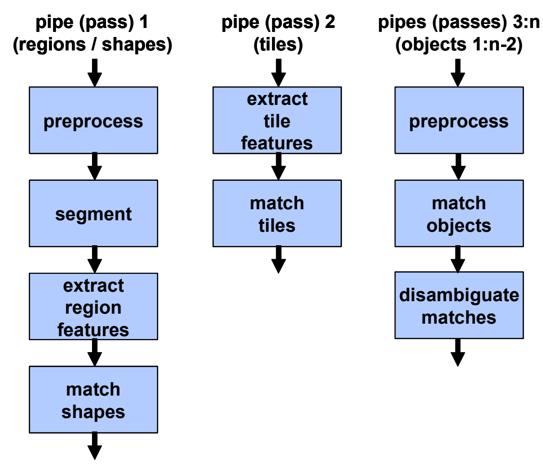

Fig.2 ICE pipeline structure.

III. DETECTION AND QUERY PROCESSING

User-interactive detection is based on the output of automated feature extraction. For each type of entity (e.g., a wooded tile, a facility, an elliptical region, etc.), a human analyst deliberately chooses a similarity detection threshold that is too low, and ICE then returns thumbnail images of potential detections sorted by figure-of-merit. The analyst then identifies a thumbnail for which most of the previous thumbnails in the list are valid detections, and most of the remaining thumbnails are false alarms. ICE will also eventually allow the analyst to identify which specific thumbnails correspond to valid detections. For a given type of entity, the analyst can currently choose from among thumbnail sorting criteria based on max, mean and cumulative similarity, for which the figures-of-merit are the max, mean and sum of similarities of entities of designated type whose centroids lie within the thumbnail.

Detected entities in the accepted thumbnails are committed to system database tables (one row contains features for one detected entity), which are suitable for standard database type queries (e.g., "Find all thumbnails from image set S containing regions of shape type A, that are within 500 pixels of wooded tiles.").

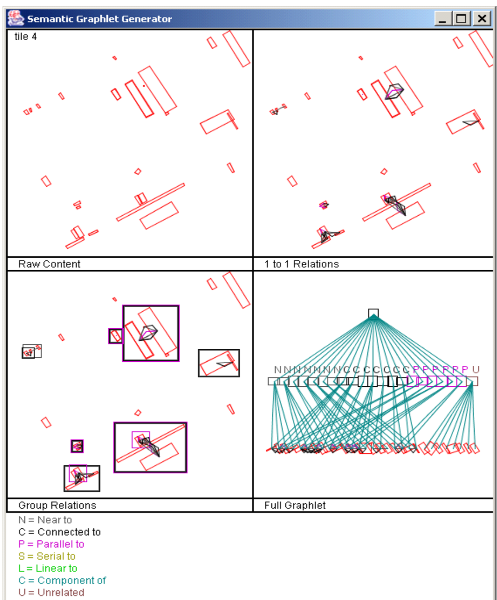

Fig.3 A graphlet constructed with content from one image block.

Detected entities are also placed into semantic graphs, which provide a convenient way to capture relationships between detected entities. Unlike relational databases, semantic graphs support queries that involve more complex relationships between detected entities (such as "Show me all areas that contain four or more parallel storage tanks, where at least one is next to a building."). ICE does not currently build semantic graphs that encompass the entire image. Images are instead broken into thumbnail-size blocks for which smaller graphs ('graphlets") are built. Graphlets contain three levels of nodes (see Fig.3). The leaf nodes (level 3) correspond to individual detected entities. The internal nodes (level 2) correspond to groupings of entities. The root node (level 1) represents the entire thumbnail with high-level "summary" information (such as the number and type of entities detected in the thumbnail, the number of groups of parallel objects, etc.). All level 2 nodes are connected to the root node. Each level 2 node is connected to the level 3 nodes that belong to the set represented by the level 2 node. Each level 3 node is connected to other level 3 nodes that it has an identified relationship with, and the links are labeled with the type of relationship (e.g., "close to", "alongside", etc.). Summary information provides a convenient mechanism for dramatically reducing the number of graphlets that must be subjected to computationally expensive full sub-graph matching.

IV. EXAMPLE

Ortho-rectified images with 1m ground sample distance from five different areas of the US were downloaded from TerraserverUSA: Aspen (Colorado), Mammoth Lakes (California), Martinez (California), Omaha (Nebraska), and White Sands (New Mexico). Each image covers roughly 6 km x 10 km (or is roughly 6000 rows x 10000 columns of pixels in size), and they collectively cover an area of roughly 300 km^2 on the ground (totaling roughly 0.3 gigapixels of imagery).

Because the images are orthorectified, a 3D object will project the same at all locations within all five images, so in this case, only its 2D signature (projection) is needed. Fig.4 shows 2D signatures for an H-shaped building and a cylindrical storage tank of particular size alongside the five TerraserverUSA image chips. Phase sensitive matching and match disambiguation required roughly 5-10 minutes per image chip per object on a 1.7GHz Pentium 4 processor.

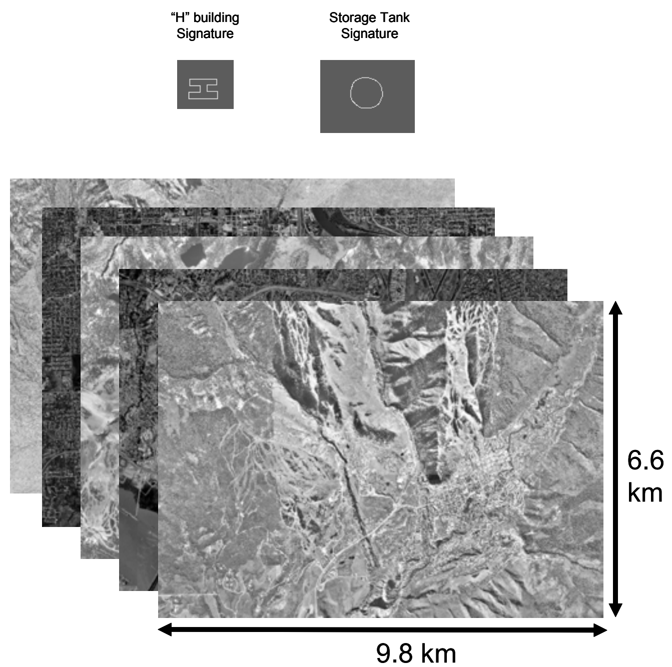

Fig.4. Signatures and images for detection example. (images courtesy of TerraserverUSA)

The ICE interactive detector was then used to present sorted thumbnails of candidate detections to a human analyst. For each object, the analyst then chose a thumbnail below which most of the thumbnails are false alarms. The results are displayed in Fig.5. The entire interactive detection session required only a few seconds – much faster than the nearly the 30 minutes it takes for an analyst to search through the image chips by hand, and for both objects, most, or all, of the objects were detected, and there were few or no false alarms. H-shaped buildings were only present in the White Sands image, whereas tank farms were present in both the Martinez and Omaha images.

Fig.5 Thumbnail hits from the interactive ICE detector for the signatures in Fig.4. (Images courtesy of TerraserverUSA)

V. SUMMARY

The Image Content Engine (ICE) has been described. It is an evolving system for providing time critical assistance to human analysts in analysis of large volumes of remotely sensed image data. ICE employs a model for computer-assisted analysis of remotely sensed images in which feature extraction and shape/object matching are performed off-line automatically, followed by user-interactive detection of discrete entities, and then user-interactive query processing to focus human attention on areas that contain collections of discrete entities that have prescribed relationships to one-another. Because feature extraction and matching are computationally expensive and there will be constraints on the amount of time available to analyze images, it will ultimately be necessary to host ICE feature extraction on parallel processing clusters. Once the features have been extracted, we have demonstrated that ICE can dramatically reduce the search time for specific objects over large volumes of imagery.

SELECTED REFERENCES

[1] I. Cox, M. Miller, T. Minka, T. Papathomas and P. Yianilos, “PicHunter, Bayesian Content-Based Image Retrieval System”, IEEE Trans. Image Proc., Vol.9, No.1, January 2000, pp.20-37.

[2] B. Zhu, “Large-Scale Content-Based Airphoto Image Digital Library”, IEEE Trans. Image Proc., Vol.9, No.1, January 2000, p.163-167.

[3] M. Hu, “Visual Pattern Recognition by Moment Invariants”, IRE Trans. Inform. Theory, Vol.8, 1962, pp.179-187.

[4] D. Paglieroni and A. Jain, “Fast Classification of Discrete Shape Contours”, Pattern Recognition, Vol.20, No.6, 1987, pp.583-598.

[5] D. Paglieroni, “Phase Sensitive Detection of 3D Objects in 2D Images”, Lawrence Livermore National Laboratory Technical Report, Summer 2004.

[6] T. Crimmins, “Geometric Filter for Reducing Speckle”, Applied Optics, Vol.24, 1985, pp.1438-1443.

[7] D. Paglieroni, “Convergent Coarseness Regulation for Segmented Images”, IGARSS04, September 2004.

[8] D. Paglieroni, “Design Considerations for Image Segmentation Quality Assessment Measures”, Pattern Recognition, Vol.37, No.8, 2004.