PERFORMANCE EVALUATION FOR AUTOMATIC FEATURE EXTRACTION

David M. McKeown, Ted Bulwinkle, Steven Cochran, Wilson Harvey, Chris McGlone,Jefferey A. Shufelt
Digital Mapping Laboratory
Computer Science Department
Carnegie Mellon University
5000 Forbes Ave
Pittsburgh, PA 15213
1-(412)-268-2626
1-(412)-268-5576 (fax)
dmm,geb,sdc,wah,jcm,js@cs.cmu.edu

Working Group II/6

KEY WORDS: feature extraction, performance evaluation, metrics

ABSTRACT

Recent years have seen significant improvements in the performance of automatic cartographic feature extraction (CFE) systems. Early systems, painstakingly tweaked to work in a very limited fashion on small images, have given way to systems which produce credible results in situations representative of production environments. While no existing automatic system is yet ready for incorporation into a production environment, significant strides have been made toward this goal. Semi-automated systems, with significant man-in-the-loop capabilities, continue to be developed by photogrammetric workstation vendors.
However, a fundamental requirement for system development, and an absolute prerequisite for production applications, is the rigorous evaluation of automated system performance. Indeed, without meaningful evaluation, we can hardly be said to be doing science. Rigorous evaluation requires the definition of a set of metrics, relevant to user requirements and meaningful in terms of expected system performance. These metrics must be generated across common, well-documented datasets which are representative of production sources and scenes.
To provide concrete examples of system evaluation techniques, this paper describes work in the Digital Mapping Laboratory on the evaluation of automated cartographic feature extraction systems. Activities include the definition and publication of metrics for several types of feature extraction systems, the generation and distribution of several large test data sets, and extensive evaluations on our CFE systems. The paper concludes with a discussion of future activities and directions in system evaluation.

1 INTRODUCTION

An important but often overlooked aspect of the development of systems for automated cartographic feature extraction is the *rigorous* evaluation of their performance. By "rigorous," we mean an evaluation process with precisely defined characteristics. It must have *clearly defined criteria*, motivated by the application requirements. The numbers it produces must be *relevant* to understanding algorithm performance, and, if motivated by a specific user application, must be relevant to the user's requirements. Performance measures must be *objective* and *quantitative*, instead of scores based on an operator's judgments when a particular building is detected or a road is correctly delineated. The system's output must be *measured against reference data* of verifiably higher quality, which in most cases means manually generated. The evaluation must be *repeatable*; given the system's output and the reference data, another laboratory must be able to reproduce the scores.

Rigorous evaluation is important for both scientific and engineering reasons, allowing us to understand the weaknesses and strengths of any particular algorithm or system and highlighting areas for improvement. Comparing the approach currently under investigation with past versions, alternative implementations, or competing systems allows the developer to focus his efforts on the most promising paths, while not pursuing dead end approaches. Characterizing system performance against a range of image and scene characteristics gives a better understanding of the true potential of an algorithm, while highlighting possible weak points.

From an engineering standpoint, we must realize that automated systems for cartographic feature extraction are close to a level of competency which would permit their adoption for production purposes. However, they will not be integrated into production systems until their output quality and economic advantages can be convincingly documented. Users must be assured that the system output meets their product standards, which can only be done by rigorous testing and documentation. From an economic standpoint, automated systems will not be adopted until users are assured that the adoption cost is economically justified; again, this must be done by documentation of algorithm efficiency and productivity.

This paper attempts to sketch the current state of evaluation research, then gives an overview of evaluation techniques used in the Digital Mapping Laboratory for a number of different feature types and systems. In conclusion, we outline what we see as the requirements for wider adoption of rigorous evaluation techniques across the CFE community.

2 CURRENT WORK ON PERFORMANCE EVALUATION

Performance evaluation of computer vision systems has become a topic of greater interest in recent years, as evidenced by recent workshops (First and Second Workshops on Empirical Evaluation Methods in Computer Vision, Workshop on Performance Characterisation and Benchmarking of Vision Systems, Evaluation and Validation of Computer Vision Algorithms) and journal special issues (PAMI, April 1999, CVIU, to appear) devoted to the topic. The CFE community has been slower to adopt rigorous evaluation methods; this may be due to many factors (Förstner, 1996), among them the expense in generating data sets and reference data, the small size and diversity of the CFE research community, and to some extent, a lack of appreciation of the requirements for evaluation.

Fortunately, this has been changing in recent years. Several recent CFE workshops have included sessions on evaluation, such as the Ascona workshops (Automatic Extraction of Man-Made Objects from Aerial and Space Images (II), 1997) and the recent ISPRS workshops in Paris (3D Geospatial Data Production: Meeting application requirements, April, 1999) and Munich (Automatic Extraction of GIS Objects from Digital Imagery, Sept. 1999), while articles on evaluation of CFE have been included in several workshops on the evaluation of general computer vision systems. ISPRS Working Groups have been responsible for the distribution of several test data sets, such as the ISPRS WG III/3 test on image understanding (Fritsch *et al.*, 1994) and the OEEPE/ISPRS WG II/8 test on performance of tie point extraction in automatic aerial triangulation (C. Heipke, 1997).

Building extraction evaluation has been advanced by the distribution of several test data sets, including the RADIUS modelboard and Ft. Hood data sets and the ISPRS Avenches data sets, among others. A summary of recent work in the evaluation of automated monocular building extraction systems is given in (Shufelt, 1999b).

The majority of work currently is directed toward semi-automated building extraction systems; several groups have implemented semi-automated systems and have published evaluation results, at some level of detail (Hsieh, 1995, Gülch *et al.*, 1998). A common weakness, however, is that these evaluations seldom include comparisons of the time and user effort required by the semi-automated system, compared to manual methods. This is partially due to the fact that exactly comparable manual systems seldom exist; however, the efficiency of semi-automated systems relative to manual ones is the crucial question and needs to be addressed. Published results also make it difficult to compare the relative merits of different semi-automated systems, given that very few tests have been run on the same datasets and that the extracted building models often have different levels of detail and attribution.

A number of laboratories are currently working on road network extraction, and have published evaluation results. McKeown and Denlinger (McKeown and Denlinger, 1988) used evaluation measures (length of tracked road and subjective evaluation) to demonstrate that cooperative extraction methods out-perform individual extraction methods. Ruskone (Ruskone, 1996) compares GIS data to automatically derived road vectors using measures for geometric accuracy, redundancy, and omissions. Heipke, et al. (Heipke *et al.*, 1997) have used our metrics, and added metrics to include measures of the accuracy of the 2D delineation.

3 PERFORMANCE EVALUATION AT THE DIGITAL MAPPING LABORATORY

3.1 Evaluation philosophy

Each type of cartographic feature has its own characteristics which must be considered in evaluating its extraction, *i.e.*, whether it is 2D or 3D, whether location, geometry, or attribution are primarily of interest, etc. In addition, differing user or application requirements may emphasize different feature aspects. Despite these differences, however, a common approach to performance evaluation can be defined and applied. The Digital Mapping Laboratory has defined such a philosophy in its work, and applied the basic techniques to a variety of cartographic feature extraction systems.

System performance evaluation at the Digital Mapping Laboratory begins with the **generation of detailed reference data for a number of different data sets**. We make significant investments in data set acquisition, registration, and documentation, and in updating existing test areas with new imagery when available. New test areas are added according to data availability, to address customer requirements, or to support new systems. We have developed a number of cartographic tools to support the generation of reference data, optimized to collect the required information. As an example of a typical reference model, Figure 1 shows a reference model constructed for the Oakland section of Pittsburgh, containing 193 buildings.

We have also invested in an extensive infrastructure for the **automated generation of evaluation metrics**; instantiating the calculation of metrics into standard re-usable packages amortizes the cost over multiple projects and makes the generation of metrics more reproducible and less onerous.

This automated metric generation facilitates the **evaluation of algorithms during development**; algorithms are re-evaluated at significant development milestones, to guard against the introduction of bugs and to quantify any performance improvement. **Evaluation results are archived** at significant system development milestones to enable long-term quantitative comparisons of system changes.

In order to facilitate community discussion and utilization of performance evaluation metrics, we are currently working on an evaluation package designed to be distributed across the user community (Bulwinkle and Shufelt, 1998, Bulwinkle

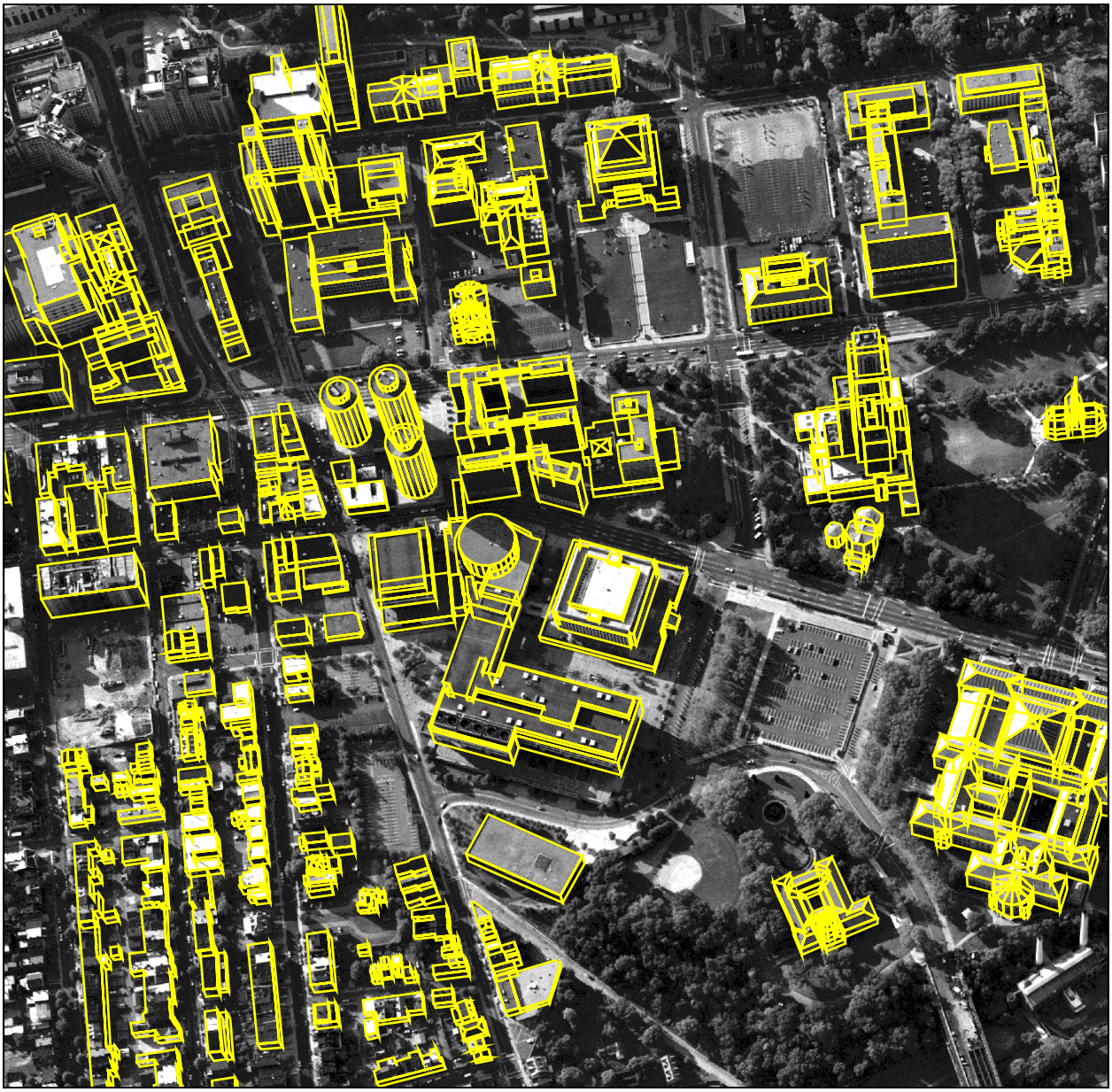


Figure 1: Oakland reference model.

et al., 1998). This package will consist of a well-documented exchange format for buildings and roads, software to automatically calculate a set of metrics, and sample imagery and reference data sets. This package is intended as an initial framework for community interaction on evaluation metrics. The software is easily extendible to include new metrics or new evaluation modalities.

Each of the following sections describes our work in performance evaluation for systems designed to extract a specific type of cartographic feature. While the specifics may vary, depending on the type of feature, we feel that these demonstrate our basic approach to evaluation. Note that results are given only to illustrate the evaluation method, and not as an indication of the performance of any given system.

3.2 Buildings

Buildings are among the most complex cartographic features to extract, due to their wide variety of complex shapes and appearances, the high probability of occlusion by surrounding objects, and the complex scenes in which they usually occur. This complexity propagates into the systems designed to extract them, making the design of relevant and efficient metrics an important issue.

A number of building extraction metrics can be and have been defined, to characterize different aspects of the process or to reflect the requirements of different user communities. For instance, building *detection*, indicating the presence of a building at a given location, is a different aspect of performance than building *delineation*, showing the boundaries of the building. Different metrics are required for each case.

Building metrics may use either an area/volume comparison or a building-by-building count. Area/volume metrics compare the system's label for each pixel or voxel to the reference data, and compute statistics based on the consistency of the labels. Alternatively, if each building hypothesis produced by the system can be related to a building in the reference data, a measure of building detection can be based on the amount of overlap achieved. Area/volume metrics are affected by building size; a system that misses small buildings but does well on large buildings will not be penalized very much. Building counting metrics weight all buildings equally, but establishing correspondence may be difficult when multiple hypotheses overlap a single building, when a number of buildings in the reference data are connected, etc.

It is assumed that semi-automated building extraction systems will generate correct building hypotheses, since the operator is guiding the process. Relevant metrics for such systems therefore involve the amount of time or effort required, either the total elapsed time or the editing time required for corrections.

The MAPSlab has implemented several building extraction systems, both semi-automated and automated. Evaluation of each has elements in common and also elements specific to the type of system, as described below.

3.2.1 Evaluating semi-automated building extraction systems (SiteCity) SiteCity is a semi–automated multi–image site modeling system, combining interactive measurement tools, image understanding techniques, and a rigorous constrained least–squares photogrammetric bundle adjustment package for site model generation. Thorough descriptions of SiteCity appear elsewhere (Hsieh, 1996a, Hsieh, 1996b).

A comprehensive evaluation of SiteCity was undertaken, as described in (Hsieh, 1995), in an effort to answer three fundamental questions:

- Does the use of automated processes introduce bias into the measurements?
- Given partially-delineated buildings structures, do the automated processes detect and delineate the buildings correctly? Do the automated processes use the operator's cues correctly?
- Is the inclusion of automated processes helpful to users? Do they reduce the user's work load and the elapsed time required?

Two distinct evaluation methodologies were applied. The Goals, Operators, Methods, and Selection (GOMS) method (John and Kieras, 1994, Card *et al.*, 1980, Rhyne and Wolf, 1993) compares the number of tasks and sub-tasks performed by the user in fully manual operation and when using the automated processes. This provides a measure of the operator's work load and the relative advantage provided by the automated processes. The second evaluation methodology was to record the elapsed time required to complete each task and sub-task, to give an estimate of the overall efficiency improvement added by the automation.

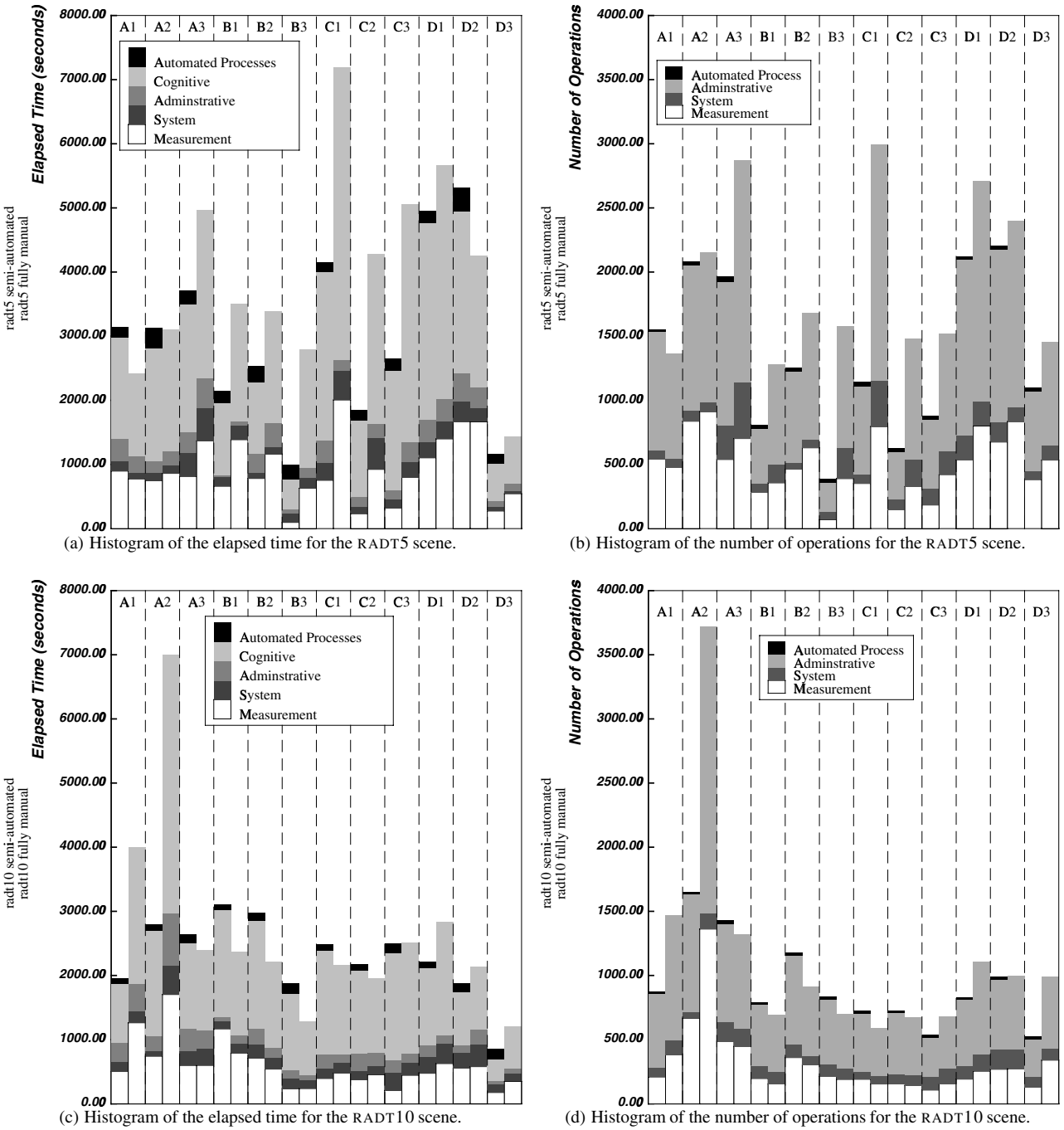
In the evaluation, twelve subjects used SiteCity to measure buildings in three test scenes using both fully manual and semi-automated modes. The order of the test scenes, as well as the order of manual and automated operation, was randomized.

To answer the three questions above, a number of experiments were performed and statistics derived, including:

- Measurement variance for manual and automated measurements.
- Vector distance between automated and manual measurements, to look for bias in measurements.
- Tests on the building hypothesis verification process to quantify its ability to discriminate between buildings and random backgrounds.
- Experiments on the effects of initial point measurement error on automated building delineation.
- Tests on the sensitivity of automated building delineation processes to hypothesis displacement errors.
- Statistics on the accuracy of automated building vertex delineation compared to manual measurements, taking into account the measurement variance of both sets of measurements.
- Counting the number of user measurement tasks to quantify user work load.
- Recording the time required for each category of user task.

Figure 2 shows histograms of the elapsed time and the number of operations for each of the 12 users (A1-D3) for two different scenes, in both manual and automated measurement modes. While the complete set of statistics will not be reproduced here, the overall conclusions were that the automated processes were no less accurate than manual measurements, and were actually more consistent. The use of automated processes reduced the elapsed time by 20% and user cost, defined by the number of tasks performed, by 12%.

SiteCity is still in use for the generation of building reference models; we use the internal instrumentation to record the amount of effort required to generate the reference models. A typical set of times, for preparation of the Oakland reference



(a) Histogram of the elapsed time for the RADT5 scene.

(b) Histogram of the number of operations for the RADT5 scene.

(c) Histogram of the elapsed time for the RADT10 scene.

(d) Histogram of the number of operations for the RADT10 scene.

Figure 2: Plot of elapsed time and number of operations for all measurements

Table 1: Elapsed times for Oakland reference model creation.

Time	% of Total	Man Days	Sec. / User Task
User	5.39	1.41	24.05
Admin	1.79	0.47	0.36
System	12.70	3.32	12.40
Cognitive	80.12	20.92	—

model, is shown in Table 1. A total of 26.12 man days were required. This was divided into four categories; *user* time, when the user was actively measuring or modifying building models, *admin* time, when the user was adjusting the view, zooming windows, etc., *system* time, when the automated processes or the photogrammetric solution were executing, and *cognitive* time, when the user was not interacting with the computer but instead examining the imagery or reference materials, planning the types of models to use, or verifying the results of the solution.

The Oakland reference model is based on measurements in six aerial images (two near-nadir and four oblique) covering

an area of approximately one square kilometer. The reference model contains 193 buildings, built from 378 volumes combined with 537 constraints. The volumes are composed of 3,795 surfaces, 17,182 edges and 34,364 points.

3.2.2 Evaluating monocular automated building extraction systems (BUILD, VHBUILD, PIVOT) Thorough and rigorous performance evaluation of automated building extraction systems has a long history at the MAPSLab. The metrics described in this section have their roots in the evaluation of fusion methods for multiple building extraction systems (Shufelt and McKeown, 1993), As our research systems were augmented to produce object space models, the metrics were extended to handle full 3D evaluation (McGlone and Shufelt, 1994, Shufelt, 1996), and deployed for comparisons of system performance as image obliquity and object density and complexity were varied (Shufelt, 1999a). A recent paper on evaluation of automated building extraction covers these topics in detail (Shufelt, 1999b); in this section, we briefly discuss the motivation for the metrics, their definitions, and how they are employed in our performance evaluation work.

Our goal in automated building extraction performance evaluation is to compare the scene models produced by an automated building extraction system for a given set of aerial imagery with the most accurate and precise scene models which can be generated from the same set of imagery. In practice, scene models which meet these requirements are compiled by manual measurements of image points, combined with photogrammetric triangulation of these points to produce 3D wireframes in object space. In our work, reference scene models were compiled with the use of SiteCity (Section 3.2.1).

The metrics we employ in our evaluation efforts have several key properties that we believe any useful set of performance metrics must possess for meaningful quantitative evaluation to take place:

- *Unbiased evaluation*: The metrics should measure performance uniformly over the entire space of possible results, or quantify the bias resulting from treating results nonlinearly. For example, a building detection measure that treats a perfect building match as equivalent to a partial building match, without quantifying the distinction, is biased.
- *Objective*: The metrics should not be tunable via thresholds, nor should human judgment play a role in determining strictly quantitative performance measures. Many existing metrics require a user to set a threshold to determine building hit/miss ratios, which introduces a subjective element into evaluation.
- *System-independent*: The metrics and the quantitative results they produce must be independent of any particular building extraction system or methodology. The metrics must not depend on specific building representations or building extraction algorithms, and must be applicable to any system generating volumetric or polygonal descriptions of scene structure.

The metrics we use for building extraction evaluation are based on classifications of pixels in image space and voxels in object space by the building extraction system's scene model and by the reference scene model. A scene model classifies each pixel in an image, and each voxel in object space, into one of two categories: object or non–object (background). Since there are two scene models in question (the one being evaluated and the reference), there are four possible categories for each pixel (or voxel):

- **true positive (TP)**: both the vision system's scene model and the reference scene model classify the pixel (voxel) as belonging to an object.
- **true negative (TN)**: both the vision system's scene model and the reference scene model classify the pixel (voxel) as belonging to the background.
- **false positive (FP)**: the vision system's scene model classifies the pixel (voxel) as belonging to an object, but the reference scene model classifies the pixel (voxel) as belonging to the background.
- **false negative (FN)**: the vision system's scene model classifies the pixel (voxel) as belonging to the background, but the reference scene model classifies the pixel (voxel) as belonging to an object.

To evaluate performance, the number of TP, TN, FP, and FN pixels and voxels are counted, and then the following metrics are computed, once for image space and once for object space:

- **building detection percentage**: $\frac{100\,TP}{TP+FN}$
- **branching factor**: $\frac{FP}{TP}$
- **quality percentage**: $\frac{100\,TP}{TP+FP+FN}$

The *building detection percentage* is the simplest metric, measuring the fraction of reference pixels which were correctly denoted as object pixels by the vision system. The *branching factor* is a measure of the degree to which a system overclassifies background pixels as object pixels. If a system never "overdelineates" the extent of any object, its branching

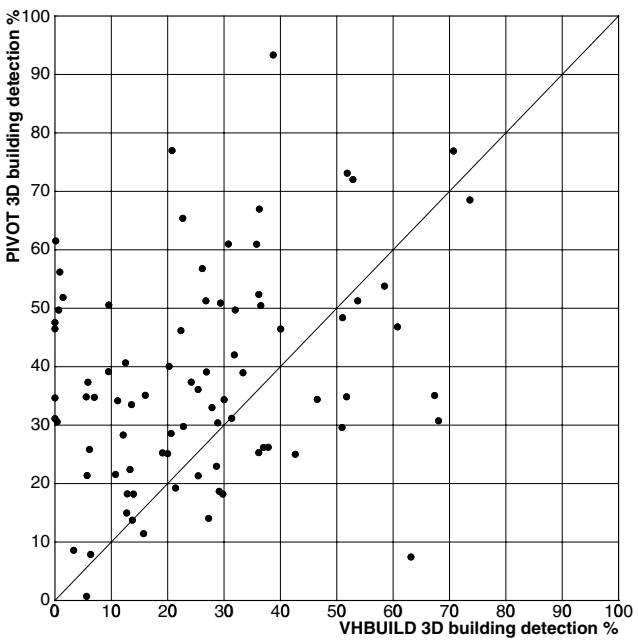


Figure 3: 3D building detection %, VHBUILD vs PIVOT.

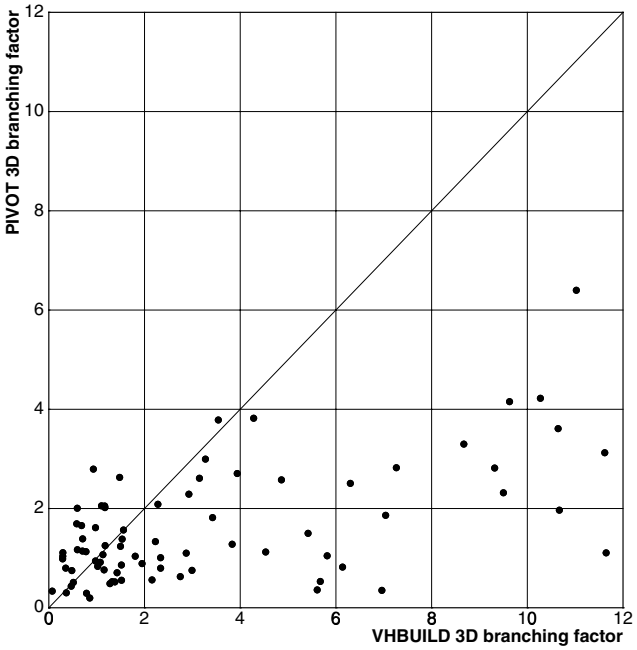


Figure 4: 3D branching factor, VHBUILD vs PIVOT.

factor would be zero, the best possible branching factor. A system with a branching factor of one would incorrectly label a background pixel as an object pixel for every object pixel it correctly detected. Finally, the *quality percentage* measures the absolute quality of the vision system's scene model. To obtain 100% quality, the vision system must correctly label every object pixel, without missing any (FN $= 0$), and without mislabeling any background pixels (FP $= 0$). In other words, the vision system must produce a perfect classification of the image with respect to the reference image to obtain a quality percentage of 100%. Implementation details for these metrics can be found elsewhere (Shufelt, 1999b).

The building detection percentage can be treated as a measure of object detection performance; the branching factor can be treated as a measure of delineation performance. The quality percentage combines aspects of both measures to summarize system performance. All three measures taken together give a clear picture of system performance with simple and unambiguous interpretations and without any subjective elements.

Figures 3 and 4 show a small part of a comprehensive building extraction evaluation using the metrics described above (Shufelt, 1999b). In this evaluation, the performance of two building extraction systems, VHBUILD (McGlone and Shufelt, 1994) and PIVOT (Shufelt, 1996, Shufelt, 1999a) were compared on 83 images of 18 test scenes. In both figures, each point represents one image.

PIVOT makes use of a more detailed model of vanishing point and shadow geometry than VHBUILD, as well as methods for robustly constraining the search space for building hypotheses. In Figure 3, the majority of the points lie on the left side of the diagonal line, indicating that PIVOT's 3D building detection performance is superior to VHBUILD's over a large sample size. Because PIVOT is able to make use of both vertical lines and shadows to estimate building height from a single image, unlike VHBUILD which only uses verticals, it tends to over-hypothesize building structure less often. Figure 4, a comparative plot of 3D branching factors, clearly illustrates this; many of the points lie well to the right of the diagonal, indicating that VHBUILD generates significantly more false positive voxels than PIVOT.

The previous example illustrates the utility of the performance evaluation metrics; they provide global measures of detection performance, quantified in an unbiased way, that permits direct system-to-system comparison. Other work has shown how these metrics can be used in conjunction with measures of image and scene complexity to analyze system performance as image and scene properties are varied (Shufelt, 1999b).

In current work, we are exploring extensions to these core metrics to support building-by-building evaluations, as a supplement to the global evaluations we already produce. We extend the basic metrics in a straightforward way. Each pixel, in addition to receiving a classification as TP, TN, FP, or FN, also receives an index to a particular building in the reference model and in the hypothesized model. This allows the totals for TP and FN to be computed for each building in the reference model. Once the totals are computed, we can easily generate building detection percentages for each building in the reference model. Given these building-by-building detection percentages, we now seek a symbolic-level measure of performance.

One proposed solution to the building-by-building detection problem defines a threshold, and calls a building detected if its individual detection percentage is greater than the threshold. This solution leaves open the choice of threshold, generally justifying a particular choice based on a particular application, which violates our notion of an objective metric. Our solution is based on the notion of a detection threshold; in conjunction with the core metrics, we can develop metrics for

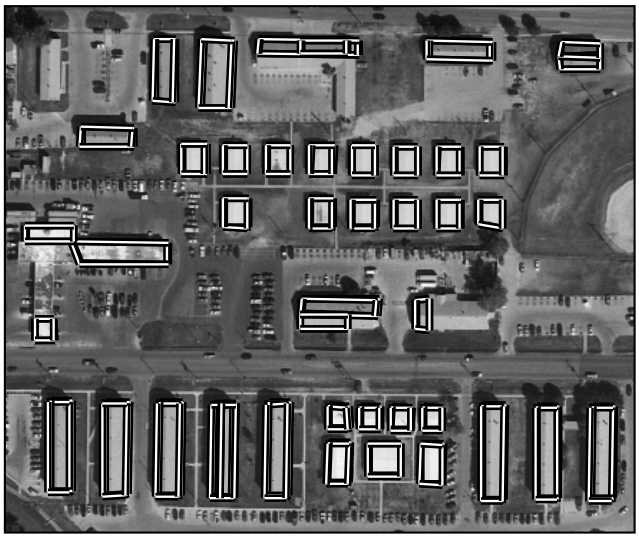


Figure 5: BUILD+SHAVE result, Fort Hood scene.

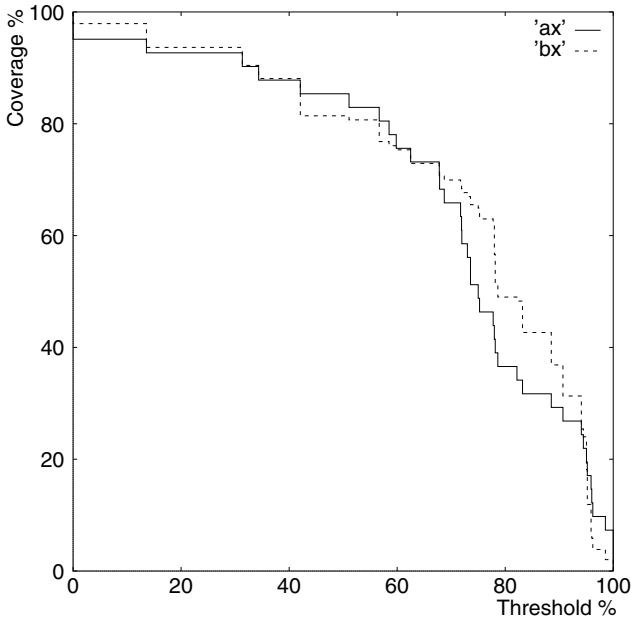


Figure 6: $a(x)$ and $b(x)$ for BUILD+SHAVE result.

computing a building-level detection percentage and a "building hit count," and quantify the bias introduced by ignoring building area/volume.

Consider plotting a 2D graph, in which the X-axis represents the detection threshold percentage, and the Y-axis represents the percentage of buildings successfully detected at that threshold, measured in terms of percentage of pixels. Let this curve, the *area–weighted detection threshold function*, be identified by $a(x)$. For any chosen threshold x, $a(x)$ tells us the percentage of buildings detected at that threshold, weighted by building area in pixels. A graph of $a(x)$ provides a visual method for performance assessment; for example, if $a(0) < 1$, we know that we have completely failed to detect some buildings. If $a(x)$ has a high value as x approaches 1, we know that we have robustly detected buildings, correctly labeling most of the building pixels. Similarly, we can define the *building–weighted detection threshold function*, which we will call $b(x)$. This threshold function. unlike its area–weighted counterpart, treats each building equally, independent of actual size. Graphical analysis of these functions can be useful for judging the behavior of a building hypothesis system. We close this section with an example of this type of analysis.

Figure 5 shows a 3D building extraction result generated by the BUILD+SHAVE system (Irvin and McKeown, 1989). Figure 6 shows the area and building-weighted detection threshold functions for this result, using SiteCity models as the reference data. $a(x)$ and $b(x)$ are similar, but not identical, showing that some bias is introduced in $b(x)$. Generally speaking, with a detection threshold of 50%, BUILD+SHAVE is able to extract 80% of the structures; performance degrades more steeply from that point. With a detection threshold of 90%, BUILD+SHAVE is only able to extract 30% of the structures. As is clear from the graph, this style of performance curve analysis allows for clear depiction of performance trends without requiring a subjective fixed threshold, as well as providing a quantitative measure of the bias introduced by treating buildings at a symbolic level rather than a pixel/voxel level.

3.2.3 Evaluating multi-image automated building extraction systems (MultiView) The MultiView system (Roux and McKeown, 1994) uses multiple images to generate 3D building hypotheses. It begins by matching corners extracted from the images, using geometric constraints to restrict the search range. The matched 3D corners are used as nodes in a 3D graph, with links between nodes corresponding to image edges. Links are weighted according to the strength and of the corresponding edge Polygonal surfaces are formed by searching the graph for cycles, based on the 2D and 3D geometry and on the strength of the edge evidence. The initial matching is done on a pair of images, with additional images added sequentially.

Evaluation of MultiView used the volumetric (voxel) metrics described above (Section 3.2.2), along with another set of metrics designed to characterize building delineation accuracy.

The delineation metrics start by associating each building hypothesis with a building in the reference data. The association requires the hypothesis and the reference buildings to be of the same type (flat roof, peaked roof, etc.), that the building footprints overlap by at least 50%, and that only one hypothesis can be matched with each reference building. In the case of multiple matches, the hypothesis with the best error score is selected.

Each reference building is described by a set of unique dimensional vectors, which describe the basic dimensions of the building; for instance, a rectangular building is specified by three dimension vectors (length, width, and height), while an L-shaped building requires seven. Statistics are calculated on the position error, the dimensional error, and the vector orientation error.

Table 2: Representative MultiView evaluation results.

Building 1	Length	Width	Height
Ground Truth	99.13	59.51	16.64

Hypothesis	Length Error	Width Error	Height Error	Position Error	Orient. Error
4 views	-1.85	-2.08	0.76	3.40	0.26°
5 views	-4.20	-1.62	0.52	3.85	0.63°
6 views	-2.70	-3.05	0.91	4.44	0.22°

(a)

Building 2	Length	Width	Height
Ground Truth	74.66	41.65	17.82

Hypothesis	Length Error	Width Error	Height Error	Position Error	Orient. Error
4 views	-0.36	-3.85	-5.83	6.10	2.33°
5 views	-1.39	-0.82	-1.39	2.89	0.94°
6 views	-1.57	0.27	-1.56	2.12	1.36°

(b)

Table 3: Data fusion experiments (Ft. Hood).

Test	Scene	2-D			3-D		
		Building Detection %	Branching Factor	Quality Percentage	Building Detection %	Branching Factor	Quality Percentage
Extraction Test	RADT5	99.94	0.21	82.43	99.78	0.54	65.05
	RADT9	99.95	0.19	84.04	99.33	0.52	65.70
	LEGO	99.77	0.11	90.13	96.08	0.32	73.33
Individual Average	RADT5	75.11	1.83	32.56	49.84	2.95	20.86
	RADT9	77.71	1.94	31.19	60.60	2.58	24.58
	LEGO	90.10	0.65	56.75	73.04	1.24	38.44
A Posteriori Fusion	RADT5	96.68	1.37	41.56	65.93	2.12	27.52
	RADT9	96.23	1.50	39.44	66.11	2.58	27.92
	LEGO	98.36	0.62	61.08	80.98	0.97	45.28
Progressive Fusion	RADT5	85.77	1.46	38.10	50.38	2.40	22.83
	RADT9	95.31	1.47	39.73	66.11	2.58	27.48
	LEGO	98.42	0.60	61.73	75.24	0.88	45.21

A complete set of evaluations was run against the RADIUS Modelboard imagery (Thornton *et al.*, 1994). Typical delineation statistics for two buildings are shown in Table 2.

Additional experiments studied the effects of changing the order in which images were added to the solution. Examination of results of these experiments led to the hypothesis that adding stereopairs of images consecutively improved the results. This provides a good illustration of the benefits of evaluation during system development; without a well-defined evaluation procedure, this effect would probably not have been noticed or could not have been confirmed.

3.3 Evaluation of stereo for building extraction

The main goal of our stereo work is to study the effects of fusing the results of our stereo matcher with the output of other feature extraction systems, in order to to develop better, more robust methods for automatic stereo matching and for cartographic feature extraction. Our current focus is on building extraction, so our current stereo evaluation uses the metrics described in Section 3.2.2, comparing building extraction results using stereo against a manually-generated reference model.

The stereo matcher that we use is the Norvelle Digital Photogrammetric Compilation Package (DPCP) stereo matcher (Norvelle, 1981, Norvelle, 1992). We have experimented with automatically generating and fusing stereo results from multiple stereo pairs to build mosaicked elevation maps of multiple flightlines. An automated process examines the elevation maps to generate building hypotheses; these may be projected back into image-space to define regions-of-interest to target other modes of feature extraction, such as the monocular building hypotheses generator PIVOT (Section 3.2.2).

Our building extraction metrics are described in Section 3.2.2. Table 3 shows the results of a set of data fusion tests in which the results of twelve stereo pairs (from four aerial images) were combined for a better and more robust elevation estimate (Cochran, 1999). The "Extraction Test" row shows the results obtained by running the building extraction algorithm on the hand-generated reference model, converted into a raster height format comparable to the stereo output. This test gives an idea of the performance of the extraction algorithm alone, without the influence of the errors present in the stereo data. The "Individual Average" row shows the average results from the twelve individual stereo runs for each test site. The last two rows show the metrics for the two fusion approaches. The a posteriori fusion approach combines the individual results in object-space, while the progressive fusion approach does the same for each level of a hierarchical stereo match, backprojecting the partial results to the image-space disparity for each stereo pair.

Both fusion approaches are more robust than, and outperform, the individual stereo runs. In addition, the fusion results generate nearly the same quantitative results as the best individual results and are much better than the average individual results.

3.4 Roads

As with our building evaluation work, the basic evaluation method for road extraction is to compare the automatically derived results against a manually compiled, high-quality reference model. For these measures, the two data sets are compared in 2D to determine overlaps. For pixel-based measures, we create masks from both the reference model data and the derived results and then compare the overlapping regions. For feature-based measures, we create a mask from the reference model data, then compare individual road features against that mask. If a feature overlaps the reference model mask by more than a given threshold (typically set to 75%), then that feature is considered a correct hypothesis. As with the building extraction performance evaluation, a confusion matrix is compiled, consisting of true-positives (TP), true-negatives (TN), false-positives (FP), and false-negatives (FN). In addition to branching factor and quality factor, several performance measures are computed:

- **Percent Complete:** $\frac{100\,TP}{TP+FN}$
- **Percent Correct:** $\frac{100\,TP}{TP+FP}$
- **Rank Distance**: $\sqrt{\frac{\%Complete^2+\%Correct^2}{2}}$
- **Percent Redundancy:** The percentage of the generated output that overlaps itself.

The *Percent Complete* is the same measure as the *building detection percentage* defined in Section 3.2.2. It measures the percentage of the reference model that is covered by the derived model and ranges from 0 to 100%, where the high values are best. The *Percent Correct* is a similar measure, but gives the percentage of the derived model covered by the reference model instead of the other way around. The *Rank Distance* is a new measure of the overall quality of the result. It measures the normalized distance (in Completeness and Correctness space) between the derived result and the reference model. Like the other two measures, it ranges from 0 to 100%, where the high values are best. Finally, the *Percent Redundancy* is useful for determining how much extra work is being done. For this measure the low values are best.

In concert, these measures provide a quantitative picture of system performance. The performance evaluation metrics we have been using are more fully described in previous papers (Harvey, 1999, Harvey, 1997). Recently, others (Heipke *et al.*, 1997) have added metrics to include measures of the accuracy of the 2D delineation.

Tables 4 and 5 show the results of evaluating two different road finding systems (Harvey, 1999), on five nadir images of Fort Hood, TX. The first, `apar`, uses anti-parallel edges to find roads while the second, `dhrf`, divides the image into overlapping tiles and uses a histogram of edge orientations within the tiles to find sets of consistent road edges. One can see that the results generated by the `dhrf` finder are much improved when compared to the results generated by the `apar` road finder:

- Branching factors, though still high, have dropped by about 30%;
- Quality factor has almost uniformly doubled;
- Rank distance has almost quadrupled;
- Completeness is in the 80–90% range;
- Correctness has increased by an average of 45%;

Redundancy has increased by an order of magnitude, mostly because the `dhrf` algorithm ensures redundancy by choosing overlapping tiles. The `dhrf` road finder improves the rate of road detection while lowering the branching factor.

Figure 7 shows the rank distance and quality of the `apar` road finder (x-axis) plotted against the rank distance and quality of the `dhrf` road finder (y-axis). This is done for each of the five test scenes. The diagonal line represents no change in performance. Points lying above the line represent performance numbers favorable to the `dhrf` finder, while points falling below the line represent numbers favorable to the `apar` finder. Though a more detailed analysis can be derived from the data in the tables, presenting the data this way allows us to quickly determine the relative quality of the systems under comparison. Figure 7 shows that the `dhrf` road finder clearly out-performs the `apar` road finder on all five Fort Hood test scenes.

3.5 Multispectral/hyperspectral classification

The Digital Mapping Laboratory's work in the utilization of multispectral (Ford *et al.*, 1993, Ford and McKeown, 1994) and hyperspectral (Ford *et al.*, 1997a, Ford *et al.*, 1998) imagery has been concerned with the generation of surface material maps and on the fusion of these surface material maps with information from other sources (McKeown *et al.*, 1999).

To support the development and evaluation of hyperspectral image classification techniques, we coordinated a data acquisition over Fort Hood, Texas, using the HYDICE sensor system and also natural color film shot by a KS–87 frame

Table 4: Performance evaluation results using the anti-parallel edge based automated road finder (no tracking).

Image	Branching Factor	Quality Factor	Rank Distance	Percent Complete	Percent Correct	Percent Redundancy
FHN711	8.55	0.08	14.76	18.06	10.47	8.98
FHN713	11.89	0.06	13.17	16.93	7.76	6.65
FHN715	6.73	0.11	16.51	19.44	12.94	9.18
FHN717	16.95	0.05	12.40	16.63	5.57	7.85
FHN719	14.86	0.05	11.90	15.60	6.31	6.91

Table 5: Performance evaluation results using the direction histogram automated road finder (no tracking).

Image	Branching Factor	Quality Factor	Rank Distance	Percent Complete	Percent Correct	Percent Redundancy
FHN711	4.82	0.17	64.27	90.07	17.17	90.68
FHN713	8.98	0.10	62.16	87.33	10.02	59.75
FHN715	5.59	0.15	64.32	89.69	15.17	62.44
FHN717	10.33	0.09	61.88	87.07	8.83	57.07
FHN719	8.85	0.10	58.94	82.74	10.15	54.77

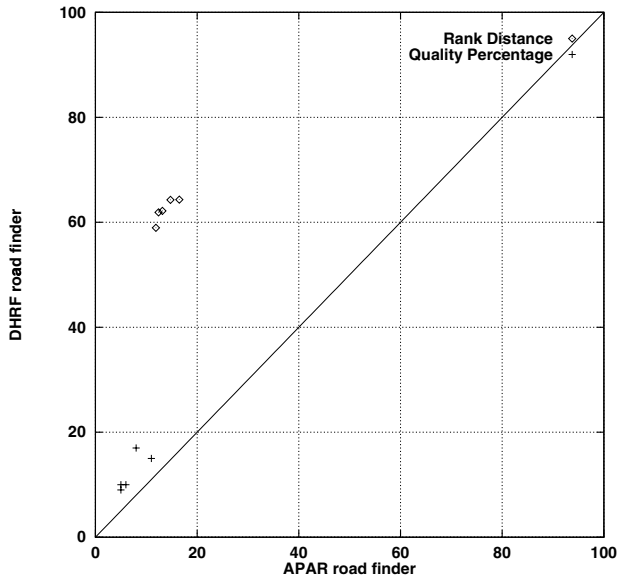


Figure 7: Comparing the results of the `apar` road finder (Table 4) to the `dhrf` road finder (Table 5).

reconnaissance camera. The spectral range of the HYDICE sensor extends from the visible to the short wave infrared (400–2500 nanometers) regions, divided into 210 channels with nominal 10 nanometer bandwidths.

Nine HYDICE flightlines, each 640 meters wide (cross-track) and 12.6 kilometer long (along-track), were flown over Fort Hood's motor pool, barrack and main complex areas. After each flightline, the HYDICE sensor was flown over and imaged a six-step (2, 4, 8, 16, 32 and 64 percent) gray scale panel, providing in-scene radiometric calibration measurements for each flightline. Prior to the start of the HYDICE flight collection, several ground spectral measurements were made for each gray level panel in an attempt to characterize its mean spectral reflectance curve. A more detailed description of the HYDICE sensor system, Fort Hood image acquisition and ground truthing activities can be found in (Ford *et al.*, 1997b).

While we have experimented with other classification methods, the majority of our work is done using supervised Gaussian maximum likelihood classification techniques. To evaluate the accuracy of the classification, we manually generate "ground truth" surface material classifications for each pixel in the area of interest. Since we typically do not have access to the scene itself, we must label pixels by examining the multispectral imagery itself in conjunction with other imagery or data sources. Any pixels which cannot be reliably classified are labeled as "unknown."

Table 6 shows a confusion matrix for a test area in the HYDICE Ft. Hood data set. As an example of the types of analysis supported by this evaluation, analysis of the error matrix shows that 56% of the classification error is associated with confusion of gravel with concrete, soil, and asphalt features. Re-examination of the training sets for gravel reveals three distinct sample populations instead of one; this multi-modal distribution violates the unimodal assumption of the

Table 6: RADT9 classification error matrix.

	REFERENCE											
TEST	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10-C14	**Row Total**	**Commission Error %**
C1	4242	2863	0	342	21	1	174	20	269	0	7932	46.5
C2	346	6922	0	77	21	83	114	66	178	0	7807	11.3
C3	0	0	116	95	0	0	61	3	36	0	311	62.7
C4	222	274	159	14043	82	0	158	158	602	0	15698	10.5
C5	2670	12971	1	821	1075	18	130	711	6033	0	24430	95.6
C6	6	15	0	9	3	460	70	35	12	0	610	24.6
C7	20	16	0	7	0	1	468	1	5	0	518	9.7
C8	966	571	23	1428	134	0	547	1467	503	0	5639	74.0
C9	183	394	6	2013	158	1	11	18	1027	0	3811	73.1
C10	13	35	0	52	0	0	16	1	122	0	239	100.0
C11	8	4	17	127	0	0	264	1	7	0	428	100.0
C12	0	0	0	0	0	0	0	0	0	0	0	*
C13	4	85	0	19	0	0	317	502	5	0	932	100.0
C14	0	0	0	0	0	0	0	0	0	0	0	*
Column Total	8680	24150	322	19033	1494	564	2330	2983	8799	0	68355	
Omission Error %	46.5	11.3	62.7	10.5	95.6	24.6	9.7	74.0	73.1	*		
Overall Accuracy = 29820 / 68355 = 43.6%												

Key	Surface material	Key	Surface material	Key	Surface material
C1	asphalt	C6	new asphalt roofing	C11	coniferous tree
C2	concrete	C7	shadow	C12	deep water
C3	deciduous tree	C8	sheet metal roofing	C13	old asphalt roofing
C4	grass	C9	soil	C14	turbid water
C5	gravel	C10	clay		

Table 7: Fine to coarse class grouping.

Coarse Surface Material	Fine Surface Material
vegetation	grass deciduous tree coniferous tree
bare earth	soil clay gravel
shadow	shadow
man-made roofing	new asphalt roofing old asphalt roofing sheet metal roofing
man-made surface	asphalt concrete
water	deep water turbid water

maximum likelihood classifier and probably contributes a significant amount of error to the classification process.

The number and types of surface material classes used in the classification are chosen to best represent the scene. We sometimes work with a reduced (or "coarse") set of classes, with semantically similar materials combined into more general classes. For instance, "grass", "deciduous tree," and "coniferous tree" classes may be combined into a "vegetation" category. Table 7 shows how the original classes were grouped into a set of of coarse classes for the RADT9 test area, and Table 8 shows the confusion matrix corresponding to this new set of classes. Using fewer classes resulted in a higher overall classification accuracy, as would be expected.

4 LESSONS LEARNED

We have learned a number of lessons from our experience.

Table 8: RADT9 coarse classification error matrix.

	REFERENCE							
TEST	man-made surface	bare earth	vegetation	man-made roofing	shadow	water	**Row Total**	**Commission Error %**
man-made surface	14373	489	419	170	288	0	15739	8.7
bare earth	16266	8415	2893	749	157	0	28480	70.5
vegetation	508	727	14557	162	483	0	16437	11.4
man-made roofing	1647	657	1479	2464	934	0	7181	65.7
shadow	36	5	7	2	468	0	518	9.7
water	0	0	0	0	0	0	0	*
Column Total	32830	10293	19355	3547	2330	0	68355	
Omission Error %	8.7	70.5	11.4	65.7	9.7	*		
Overall Accuracy = 40277 / 68355 = 58.9%								

- *The limitations of visual evaluation and comparison of results.* A standard system development technique is to run a system on one image and to compare the results to memories of earlier results. A slightly more sophisticated method is to visually compare current results to pictures of earlier results; this is all too often the only method used in research papers. Visual inspection may give valuable insights into overall system performance; however, accurately judging relative performance in two or more similar cases is difficult, since it is susceptible to the developer's wishful thinking and other distractions.

- *The need to understand the impact of scene complexity.* Building extraction is influenced both by the building complexity and the scene complexity—correctly delineating an isolated building is much easier than extracting the same building in a crowded urban area. This is partly due to imaging effects such as occlusion and shadowing and also to the effects of increased clutter on the search spaces of automated algorithms. This clutter increases the number of false hypotheses formed and evaluated and may mask correct hypotheses. Some attempts have been made to formulate metrics to describe scene complexity (Shufelt, 1999b) but at this point complexity descriptions are mostly qualitative.

- *The cost of obtaining test data sets and constructing high-quality reference data.* Test data processing involves a large amount of detailed labor, the exact opposite of the work most researchers like to perform. The MAPSlab tries to operate on a "share-the-pain" basis and spread data generation tasks across all users. We also use undergraduates working on a part-time basis, although consistency and quality control can be a problem. There is no inexpensive solution to reference data generation.

- *The utility of reference data sets at varying levels of detail.* The difference in cost between collecting highly-detailed reference data and fairly coarse data is, in most cases, small; therefore, reference data should be collected at the highest-practical level of detail, then simplified as required for any particular application.

- *Multiple applications of reference data.* A single reference data set, compiled with high levels of detail and attribution, can support a number of different evaluation modalities. Statistics can be generated for building localization, geometry, volume, height, etc., for monocular or multiple-image systems. The same data set, rasterized into a height representation, is useful for stereo evaluation. Availability of 3D building models makes the evaluation of hyperspectral imagery more accurate, since building roofs and walls can be correctly projected into the image.

5 REQUIREMENTS FOR COMMUNITY ADOPTION OF EVALUATION

There is no question that algorithms and systems are at a state of development where meaningful evaluation is needed. Rigorous evaluation techniques have been researched, developed, and demonstrated. The user community is ready and even eager to adopt automated techniques, if their utility can be proven. What is required for the research community to adopt consistent, rigorous, community-wide evaluation techniques? We feel that the *mechanisms* to enable evaluation must be widely available and that *motivation* must be provided for researchers to perform the "grunt work" of evaluation.

5.1 Evaluation mechanisms

Before evaluation will be widely practiced across the community, a solid infrastructure must be established. At a minimum, this infrastructure must include:

Common evaluation datasets, shared across the community. Valid comparisons can be done only if systems are run on same data sets and compared to the same reference data. While we assume and hope that test datasets are representative samples of real-world scenes, the number of variables involved means that the results of systems run on two different test areas can not be directly compared.

The construction of a good reference dataset is a very expensive procedure. Once the imagery and associated data is acquired and registered, the reference data must be generated. If the dataset is to be distributed, it must be documented and the data packaged in the distribution format. This process involves a significant investment of effort and requires a high level of expertise to properly register the images and generate high-accuracy reference data. The high cost deters many groups from generating good data sets, while others can afford to prepare only a few. The production cost must be amortized across a larger user base to make evaluation less costly. As an alternative, data sets could be provided by sponsoring agencies, either directly from their own production shops or contracted out to commercial mapping companies.

Common reference data formats. Communication of algorithm results and the transfer of reference data among laboratories obviously requires a common format. This format should support the calculation of the specified metrics easily; it should also be simple to convert into from laboratories' own internal formats.

Well-defined and accepted evaluation metrics. The community must agree on a set of well-defined evaluation metrics, to be published and implemented in freely-distributed software. This agreement could be imposed in a top-down manner, if it is imposed by a funding agency or sponsor, or in a cooperative manner by an ISPRS working group or an ad hoc group of laboratories.

Software to implement replication of results. To ensure that metrics are calculated consistently, a software package (with source code) must be distributed to each participating laboratory. This also relieves each laboratory from the expense of re-implementing the metrics.

Ability to extend metrics as required. The evaluation process will continue to evolve as more experience is gained, as new user requirements emerge, and as new data sources come into use. In addition, new systems may have characteristics which require new metrics to fully understand. The evaluation package must therefore be easily extensible to incorporate new metrics.

5.2 Evaluation motivations

Ideally, scientific rigor alone would be sufficient motivation for researchers to perform good evaluations: unfortunately, additional motivational factors are often needed. Many of these factors require organization and cooperation by the scientific research community. They also require collaboration with research sponsors as well as end users. Some of the most important motivations for participation include:

Sponsor involvement and evaluation requirements. Research programs must explicitly reference evaluation requirements, including procedures, metrics, and publication. Increasingly, the goals of sponsored research are becoming more pragmatic but without much thought as to how to evaluate research results.

Publication requirements. Until journal editors and reviewers require the appropriate evaluation of work described in submitted papers, there is little motivation for this work to be performed. This has improved over the last few years by requiring performance evaluation sections in submitted papers, but a community-wide source of testing data will allow greater consistency for comparisons between alternative research approaches.

Community expectation of willingness to share datasets and software. In physical and natural sciences, particularly chemistry, physics and biology, replication of results via replication of experimental process is the absolute norm. This is achieved by sharing source data, material samples, and detailed documentation of the experimental procedure. In computer science this is more difficult since the software artifacts that represent the research are not often shared. The reasons for this are outside the scope of this brief paper, but they include the fact that software represents a large investment over a long period of time for any research group and is generally the basis for future successful funding proposals. In spite of this, replication can be achieved using a common test data set and a common set of expected results, as suggested in this paper.

Guaranteed fair evaluations. Evaluation implies a competition, which may have implications for future research funding and indeed the existence of the research group. In order to be fair, the evaluations must use public standards and be implemented in publicly available software. If each researcher has the software and reference data, evaluation can be performed privately within the laboratory, then results published and publicly reported when the researcher feels they are ready. Since the evaluation can be repeated by others, the motivation to "cheat" is reduced. Additionally, evaluations should be performed at a level appropriate to the algorithm's stage of development. Basic research results implemented as a proof-of-concept can be evaluated differently than mature systems aimed at user applications. For example, a research algorithm might be evaluated more in terms of diagnostic and detection statistics than a mature system, which would necessarily be concerned with overall productivity and accuracy.

6 CONCLUSIONS

Before automated cartographic feature extraction will be taken seriously as either a science or an engineering field, rigorous performance evaluation must become a standard part of any CFE system development effort and an absolute expectation among the CFE community. Recent progress has been encouraging; for this to continue, extensive investments must be made in evaluation infrastructure, both for datasets and evaluation software, and internal and external motivations must be strengthened.

Implementation of these proposals will require real work, within the context of ISPRS, various funding agencies, and between research groups working in the area of automatic cartographic feature extraction. The ultimate motivation for engaging scientific replication in our field is its survival as a viable topic for research.

REFERENCES

Automatic Extraction of Man-Made Objects from Aerial and Space Images (II), 1997.

Bowyer, K. W. and Phillips, P. J. (eds), 1998. Empirical evaluation techniques in computer vision. IEEE Computer Society.

Bulwinkle, G. and Shufelt, J., 1998. A building model evaluation suite using the CMU site exchange format. Technical Report CMU–CS–98–133, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213.

Bulwinkle, G., Cochran, S., McGlone, J. C., McKeown, D. and Shufelt, J., 1998. Robust exchange of cartographic models for buildings and roads: The CMU MAPSLab site exchange format. Technical Report CMU–CS–98–134, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213.

C. Heipke, e., 1997. Special issue: Automatic image orientation. ISPRS Journal of Photogrammetry and Remote Sensing.

Card, S. K., Moran, T. P. and Newell, A., 1980. Computer text-editing: An information-processing analysis of a routine cognitive skill. Cognitive Psychology 12, pp. 32–74.

Cochran, S. D., 1999. Qualitative and quantitative evaluation of stereo/stereo fusion experiments. Bulletin de la Société Française de Photogrammétrie et Télédétection n. 153(1999–1), pp. 49–51. Colloque « Production de Données Géographiques 3D : vers le Respect des Contraintes Applicatives ».

Ford, S. and McKeown, D., 1994. Performance evaluation of multispectral analysis for surface material classification. In: Proceedings of the International Geoscience and Remote Sensing Symposium, IGARSS'94, Pasadena, California, pp. 2112–2116.

Ford, S. J., Hampshire, J. B. and McKeown, Jr., D. M., 1993. Performance evaluation of multispectral analysis for surface material classification. In: Proceedings of the DARPA Image Understanding Workshop, Defense Advanced Research Projects Agency, Morgan Kaufmann Publishers, Inc., Washington, D.C., pp. 421–435.

Ford, S. J., Kalp, D., McGlone, J. C. and McKeown, Jr., D. M., 1997a. Preliminary results on the analysis of HYDICE data for information fusion in cartographic feature extraction. Technical Report CMU–CS–97–116, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213.

Ford, S. J., Kalp, D., McGlone, J. C. and McKeown, Jr., D. M., 1997b. Preliminary results on the analysis of HYDICE data for information fusion in cartographic feature extraction. In: Proceedings of the SPIE: Integrating Photogrammetric Techniques with Scene Analysis and Machine Vision III, Vol. 3072, pp. 67–86. Also available as Technical Report CMU–CS–97–116, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213.

Ford, S. J., McGlone, J. C., Cochran, S. D., Shufelt, J. A., Harvey, W. A. and McKeown, Jr., D. M., 1998. Analysis of HYDICE data for information fusion in cartographic feature extraction. In: Proceedings of the International Geoscience and Remote Sensing Symposium, Vol. 5, Seattle, Washington, pp. 2702–2706.

Förstner, W., 1996. 10 pros and cons against performance characterisation of vision algorithms. In: Performance Characteristics of Vision Algorithms, Cambridge.

Fritsch, D., Sester, M. and Schenk, T., 1994. Test on image understanding. In: Proceedings: ISPRS Commission III Symposium on Spatial Information from Digital Photogrammetry and Computer Vision, Volume 30, Part 3/1, Munich, Germany, pp. 243–248.

Gruen, A. and Li, H., 1997. Linear feature extraction with 3-D LSB-snakes. In: Automatic Extraction of Man-Made Objects from Aerial and Space Images (II), Birkhäuser Verlag, pp. 287–298.

Gülch, E., Müller, H., Läbe, T. and Ragia, L., 1998. On the performance of semi-automated building extraction. In: International Archives of Photogrammetry and Remote Sensing, Vol. XXXII(3), pp. 331–338.

Harvey, W., 1997. CMU Road Extraction Test Results. (Slides presented at Terrain Week '97 in San Antonio, Texas.) Retrieved 21 February, 1997 from the World Wide Web: http://www.maps.cs.cmu.edu/rcvw/terrainweek97/roads/tw97-roadeval.ROOT.html.

Harvey, W., 1999. Performance evaluation for road extraction. Bulletin de la Société Française de Photogrammétrie et Télédétection n. 153(1999–1), pp. 79–87. Colloque « Production de Données Géographiques 3D : vers le Respect des Contraintes Applicatives ».

Heipke, C., Mayer, H., Wiedemann, C. and Jamet, O., 1997. Evaluation of automatic road extraction. In: International Archives of Photogrammetry and Remote Sensing, Vol. **32**3–4W2, pp. 151–160.

Hsieh, Y., 1995. Design and evaluation of a semi-automated site modeling system. Technical Report CMU–CS–95–195, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213.

Hsieh, Y., 1996a. Design and evaluation of a semi-automated site modeling system. In: Proceedings of the ARPA Image Understanding Workshop, Advanced Research Projects Agency, Morgan Kaufmann Publishers, Inc., Palm Springs, California, pp. 435–459.

Hsieh, Y., 1996b. Sitecity: A semi-automated site modelling system. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, California, pp. 499–506.

Irvin, R. B. and McKeown, D. M., 1989. Methods for exploiting the relationship between buildings and their shadows in aerial imagery. IEEE Transactions on Systems, Man & Cybernetics 19(6), pp. 1564–1575. Also available as Technical Report CMU–CS–88–200, Computer Science Department, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213.

John, B. and Kieras, D. E., 1994. The GOMS family of analysis techniques: Tools for design and evaluation. Technical Report CMU-CS-94-181, Computer Science Department, Carnegie Mellon University.

McGlone, J. C. and Shufelt, J. A., 1994. Projective and object space geometry for monocular building extraction. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Seattle, Washington, pp. 54–61.

McKeown, D. M. and Denlinger, J. L., 1988. Cooperative methods for road tracking in aerial imagery. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Ann Arbor, Michigan, pp. 662–672.

McKeown, Jr., D. M., Cochran, S. D., Ford, S. J., McGlone, J. C., , Shufelt, J. A. and Yocum, D. A., 1999. Fusion of HYDICE hyperspectral data with panchromatic imagery for cartographic feature extraction. IEEE Transactions on Geoscience and Remote Sensing 37(3), pp. 1261–1277. Special Issue on Data Fusion.

Norvelle, F. R., 1981. Interactive digital correlation techniques for automatic compilation of elevation data. Technical Report No. ETL–0272, U.S. Army Engineer Topographic Laboratories, Fort Belvoir, Virginia 22060.

Norvelle, F. R., 1992. Stereo correlation: Window shaping and DEM corrections. Photogrammetric Engineering and Remote Sensing 58(1), pp. 111–115.

Rhyne, J. R. and Wolf, C. G., 1993. Recognition-based user interfaces. In: Advances in Human Computer Interaction, Vol. 4, Ablex Publishing Corporation, Norwood, New Jersey, pp. 191–250.

Roux, M. and McKeown, Jr., D. M., 1994. Feature matching for building extraction from multiple views. In: Proceedings of the ARPA Image Understanding Workshop, Advanced Research Projects Agency, Morgan Kaufmann Publishers, Inc., Monterey, California, pp. 331–349.

Ruskone, R., 1996. Road Network Automatic Extraction by Local Context Interpretation: Application to the Production of Cartographic Data. Ph.D. thesis, Universite' Marne-La-Valle'e, France.

Shufelt, J., 1996. Exploiting photogrammetric methods for building extraction in aerial images. In: International Archives of Photogrammetry and Remote Sensing, Vol. XXXI, B6, S, pp. 74–79.

Shufelt, J. A., 1999a. Geometric Constraints for Object Detection and Delineation. Kluwer International Series in Engineering and Computer Science, Vol. SECS 530, Kluwer Academic Publishers, Boston.

Shufelt, J. A., 1999b. Performance evaluation and analysis of monocular building extraction from aerial imagery. IEEE Transactions on Pattern Analysis and Machine Intelligence 21(4), pp. 311–326. Special Section on Empirical Evaluation of Computer Vision Algorithms.

Shufelt, J. A. and McKeown, D. M., 1993. Fusion of monocular cues to detect man-made structures in aerial imagery. Computer Vision, Graphics, and Image Processing 57(3), pp. 307–330.

Thornton, K., Nadadur, D., Ramesh, V., Liu, X., Zhang, X., Bedekar, A. and Haralick, R., 1994. Groundtruthing the RADIUS model-board imagery. In: Proceedings of the ARPA Image Understanding Workshop, pp. 319–329.

Vasudevan, S., Cannon, R. and Bezdek, J., 1988. Heuristics for intermediate level road finding algorithms. Computer Vision, Graphics, and Image Processing 44, pp. 175–190.