International Journal on Document Analysis and Recognition manuscript No.
(will be inserted by the editor)

A Local Evaluation of Vectorized Documents by means of Polygon Assignments and Matching.

R. Raveaux, J. C. Burie and J. M. Ogier
e-mail: romain.raveaux01@univ-lr.fr

L3I laboratory, University of La Rochelle, av M. Crépeau, 17042 La Rochelle Cedex 1, France

The date of receipt and acceptance will be inserted by the editor

Abstract. This paper presents a benchmark for evaluating the Raster to Vector conversion systems. The benchmark is designed for evaluating the performance of graphics recognition systems on images that contain polygons (solid) within the images. Our contribution is two-fold, an object mapping algorithm to spatially locate errors within the drawing, and then a cycle graph matching distance that indicates the accuracy of the polygonal approximation. The performance incorporates many aspects and factors based on uniform units while the method remains non-rigid (thresholdless). This benchmark gives a scientific comparison at polygon level of coherency and uses practical performance evaluation methods that can be applied to complete polygonization systems.

A system dedicated to cadastral map vectorization was evaluated under this benchmark and its performance results are presented in this paper. By stress testing a given system, we demonstrate that our protocol can reveal strengths and weaknesses of a system. The behavior of our set of indices was analyzed when increasing image degradation. We hope that this benchmark will help assessing the state of the art in graphics recognition and current vectorization technologies.

Key words: Vectorized Object Comparison – Polygon Assignment – Polygonization Accuracy – Performance evaluation – Polygon detection quality – Graphics recognition – Machine drawing understanding system

1 Introduction

In this paper, the question of performance evaluation is discussed. A bird's-eye view methodology is adopted starting from the generic idea about evaluation of vectorization and the main principles of the proposed approach. Then, the introduction turns into a deeper discussion explaining in details both related works and the description of the proposed method.

Correspondence to: M. Raveaux

1.1 Bird's-eye view of vectorization evaluation

Driven by the need to convert a large number of hard copy engineering drawings into CAD files, raster to vector conversion has been a field of intense research for the last four decades. In addition to research prototypes in several academic and industrial research centers, several commercial software products are currently available to assist users in converting raster images to vector (CAD) files. However, the process of selecting the right software for a given vectorization task is still a difficult one. Although trade magazines have published surveys of the functionality and ease of use of vectorization products [1], a scientific, well designed, comparison of the auto-vectorization capability of the products was still required.

Responding to this need, the International Association for Pattern Recognition's technical committee on graphics recognition (IAPR TC10) organized the series of graphics recognition contests. The first contest, held at the GREC'95 workshop in University Park, PA, focused on dashed line detection [2], [3], [4]. The second contest, held at the GREC'97 workshop in Nancy, France, attempted to evaluate complete (automatic) raster to vector conversion systems [5], [6], [7]. The third contest, held off-line associated with the GREC'99 workshop in Jaipur, India, also aimed to evaluate complete (automatic) raster to vector conversion systems. These contests tested the abilities of participating algorithms / systems to detect segments and arcs from raster images. They adopted a set of performance metrics based on the published line detection performance evaluation protocol [8] to evaluate and compare the participating algorithms / systems on-line at the workshop site with test data of different quality and complexity. Pre-contest training images and the performance evaluation software were provided before the contests, so prospective participants could try their systems and improve them for optimal performance. Test images could be synthesized and/or real scanned images.

Performance evaluation and benchmarking have been gaining acceptance in all areas of computer vision and so in the graphics recognition field of science.

Early work on this topic was carried out to evaluate performance of thinning algorithms. [9] was the first to propose a general approach for performance evaluation of image analysis, with thinning taken as a case in point. Evaluation and comparison of thinning algorithms have also been performed by [10], [11], [12] and [13]. Some of these evaluation and comparison works were carried out from the viewpoint of OCR, while the work of [12] is domain independent. Although thinning may also be employed as preprocessing of line detection, the latter has different characteristics and therefore requires a different evaluation protocol.

Vectorization and other line detection techniques have been developed to convert images of line drawings in various domains from pixels to vector form (e.g., [14], [15], [16]) and a number of methods and systems have been proposed and implemented (e.g., [17], [18], [19], [20]). Objective evaluations and quantitative comparisons among the different shape detection algorithms are available thanks to protocols issued from GREC contests [21], [22], [23] that provide quantitative measurements.

Performance evaluation of vectorization and line detection has been reported by [3], [24], [8] and [5]. Kong et al. [3] propose a quantitative method for evaluating the recognition of dashed lines. Hori and Doermann [24] propose a quantitative performance measurement methodology for task-specific raster to vector conversion. Wenyin and Dori [8] present a protocol for evaluating the recognition of straight and circular lines. Phillips and Chhabra [5] define a methodology for evaluating graphics recognition systems operating on images that contain various objects such as straight lines and text blocks. All of these methods are limited in their applicability and are discussed in the next subsection.

In this article, we present a benchmark designed for evaluating the performance of graphics recognition systems on images that contain occlusions within the images. Most engineering documents could be approximated by polygons and our study focuses on these particular primitives. For instance, parcels into a cadastral map are well modeled by polygons. Accurate and efficient vectorization of line drawings is essential for any higher level processing in document analysis and recognition systems. In spite of the prevalence of vectorization methods, no standard for their performance evaluation protocol exists at a polygon level. All prior works focused on a lower level of consistency (arcs and segments). We propose a protocol for evaluating polygon extraction to help compare, select, improve, and even design object detection algorithms to be incorporated into drawing recognition and understanding systems. The protocol can be seen as an extension to polygon level of related approaches by proposing an evaluation which is closer to the user requirements (i.e. at a semantic level). This new viewpoint on the problem involves two local dissimilarity measures for estimating polygon detection and approximation quality.

1.2 Related work

Kong et al. [3] have developed a protocol and a system for systematically evaluating the performance of line detection algorithms, mainly for dashed-line detection algorithms. They define the overlap criteria of the match between a ground truth and a detected line based on the angle and the distance between them, and the partial overlap is also considered. They do not allow for fragmentation of detected lines. They use several arbitrary and rigid thresholds, for example, the angle should be less than 3° and the distance between two lines less than 5 cells.

Hori and Doermann [24] instantiate and extend Haralick's framework for performance characterization in image analysis [9], in an application-dependent manner, for measuring the performance of raster to vector conversion algorithms. They provide a set of metrics (evaluation contents) which is specifically geared to vectorization of mechanical engineering drawings. The "applications" addressed in the work are thinning, medial line finding, and line fitting all low-level techniques that do not completely constitute vectorization. It is hard to extend the work to evaluate a complete vectorization system. Hori and Doermann's protocol does not distinguish between detection rate and false alarm rate. It does not include an overall evaluation metric. It does not allow for fragmentation of detected lines. The metrics for line evaluation are given in several nonuniform units. It uses length ratio, deviation, and count ratio to evaluate the line length detection, line location detection, and line quantity detection, respectively. There is lack of an overall evaluation metric which provides an overall combined performance evaluation of the algorithm under consideration.

Wenyin and Dori [8] propose performance evaluation indices for straight and circular line detection. Detection and false alarm rates are defined at both the pixel level and the vector level. Use of pixel level performance indices (measures of shape preservation) is not completely appropriate for real images that contain severe distortions such as warping and/or other defects introduced in the hard copy drawing and/or defects generated by the scanning/imaging system. On such images, attempts to obtain a high pixel recovery index would unnecessarily require the detected vectors to be true to the distorted shape of the imaged lines, thereby making the detected lines fragmented. For such images, the pixel recovery index needs to be assigned less weight than the vector recovery index. However, there is no way to predetermine the right relative weights for the pixel and vector recovery indices.

Phillips and Chhabra [5] present the task of evaluation from the opposite angle. They do not look at the complexity of the entities to be recognized. Instead, in their view, the true measure of performance has to be goal directed. The goal of line drawing recognition is to convert a paper copy or a raster image of a line drawing into a useful form (such as a vector CAD file). How well a graphics recognition system works should be measured by how much manual effort is required to correct

the mistakes made by the system, not by how well it recognizes difficult shapes. The goal of the evaluation is to measure the cost of postprocessing operations that are necessary to correct the vectorization mistakes. EditCost is the cost estimate for human post-editing effort to clean-up the recognition result.

Based on this synthesis of performance evaluation systems, one can observe that most of these methods remain at a very low level of analysis of the information (vector level), while user requirements often concern high-level analysis. From the related work which focuses on low-level primitives (segments, arcs), we extend the global concept of performance evaluation of vectorized documents to polygon level. Herein, we present our work, a recovery index which combines a local overlapping metric at polygon level when data are closer to the semantic level and a matching distance for evaluating the polygonal approximation correctness in terms of edit operations.

1.3 Our approach

[8] and [5] are well-suited tools to tackle the performance evaluation problem of vectorized documents. However, to be more realistic and closer to objects handled by humans, [8] and [5] also underlined the need to consider more complex structures or domain-specific objects into the assessment process. For instance, in [5], Dr. Chhabra reported as a shortcoming that "The detection of polylines, polygons, objects, symbols, etc. was not tested". A step in this direction is to address the problem under the prism of grouping of vectors. Unfortunately, prior algorithms cannot be easily modified to reach higher level objects since no match was attempted between solid entities. In fact, if in the case of low-level primitives the matching can be easily reduced to an overlapping criterion; for more complex elements the question is more ambitious. Due to fragmentation phenomena introduced by R2V tools, an entity of the ground-truth can possibly be related to many elements of the auto-vectorized version of the document. Solving this ambiguity requires complex matching algorithms that are not provided by prior works because the underlying problem does not exist at a low level of analysis. Rather than consider polygon fragmentation and combination as being simply wrong, and only allow the best match with the maximum overlap, we address the question in an optimal manner to find best polygon assignments. As a consequence, to address the performance evaluation problem at polygon level, we need to provide a robust object matching. In our our approach, this major phase is carried out by a combinatorial framework to perform polygon assignments. Secondly, starting from the original idea of *EditCost* explained by Phillips and Chhabra, the cost estimate for human post-editing effort to clean-up the recognition result, we propose the use of a graph matching. This paradigm provides more than a value in $\mathbb{R}$, it reveals the sequence of corrections to be made to transform a set of connected line segments into another.

Through the reading of the literature, on the topic of performance evaluation of document image algorithms, we took into account comments and limitations of former protocols to detect five desired points:

1. To consider object fragmentation
2. To provide indices in uniform units
3. A generic and a domain-independent protocol
4. An overall evaluation metric
5. To evaluate how much manual effort is required to correct mistakes made by the system

Our proposal fulfills these five points: (1) A polygon assignment method and a graph matching algorithm tackle both polygon and line fragmentation problems; (2) Our two indices are bounded between 0 and 1; (3) No assumptions as to the kind of documents are made by our protocol, the only constraint is that the document must contain polygons; (4) An overall metric is provided by linear combination of the two proposed indices; (5) The EditCost representative of the manual labor to be made to correct a document is envisaged through the graph matching question in terms of basic edit operations (addition, deletion, substitution). Furthermore, working at polygon level offers many advantages. It makes the spotting of errors easier, there are much less polygons than vectors in a drawing, so the visualization of mistakes is pretty fast. This point is very important for industrial systems, since it permits to reduce the user correction time, by helping him to focus on errors directly. Furthermore, this facilitates the study of large samples of documents and new error categorizations may arise. Addressing the question from another point of view can help developers to improve and design R2V software.

```
We can't solve problems by using the same kind
of thinking we used when we created them.
(Albert Einstein)
```

We propose here a novel and optimal object matching for polygon comparison from a viewpoint that differs from prior works. We consider the coherency of the document at a polygon level. Our polygonization evaluation is based on a polygon mapping constrained by the topological information. While this measure appreciates the quality of the polygon overlapping, a cycle graph matching takes a closer look at a lower level of information: the segment layout within the polygons. In this way, we express the consistency of the drawing from a polygon point of view.

The smallest item that can be found in a engineering drawing is the segment.

Definition 1. *(Segment) In geometry, a line segment is a part of a line that is bounded by two end points, and contains every point on the line between its end points.*

This object is considerably versatile, the number of line segments present in a wire drawing can be significantly impacted by the vectorization algorithm due to the noise that occurred in the original image of documents (Noise due to storage conditions, digitization steps). On the opposite, we decide to investigate a more consistent and reliable object called polygon.

Definition 2. *(Polygon) In geometry, a polygon is traditionally a plane figure that is bounded by a closed path or circuit, composed of a finite sequence of straight line segments (i.e., by a closed polygonal chain). These segments are called its edges or sides, and the points where two edges meet are the polygon's vertices or corners. A polygon is a 2-dimensional example of the more general polytope in any number of dimensions.*

The polygons are formed by running a cycle detection algorithm on the heap of segments that composed the drawing. Invented in the late 1960s, Floyd's cycle-finding algorithm [25] is a pointer algorithm that uses only two pointers, which move through the sequence of points at different speeds. This polygon layer is more reliable and so it provides a better foundation to build a dissimilarity measure on top of it. A conventional way of defining measures of dissimilarity between complex objects (maps, drawing obtained by vectorization) is to base the measure on the quantity of shared terms. Between two complex objects o_1 and o_2, the aim is to find the matching coefficient mc, which is based on the number of shared terms. The polygon organization of a document is a good viewpoint, more stable and less subject to variations than the segment layer. In the mean time, it represents a complimentary view of the problem.

Polygonized elements issued from a raster to vector conversion method are assigned and measured up to a manually vectorized Ground Truth. The assignment problem is one of the fundamental combinatorial optimization problems in the branch of optimization or operations research in mathematics. It consists of finding a maximum weight matching in a weighted bipartite graph.

In its proposed form, the problem is as follows:

- Let D_{GT}, D_{CG} be a Ground Truth document and a Computer Generated document, respectively.
- There are $|D_{CG}|$ number of polygons in D_{CG} and $|D_{GT}|$ number of polygons in D_{GT}. Any polygons (P_{CG}) from D_{CG} can be assigned to any polygons (P_{CG}) of D_{CG}, while incurring some cost that may vary depending on the P_{CG}-P_{GT} assignment. It is required to map all polygons by assigning exactly one P_{CG} to each P_{GT} in such a way that the total cost of the assignment is minimized. This matching cost is directly linked to the cost function that measures the similarity between polygons.

Our combinatorial framework cuts down the algorithmic complexity to an $O(n^3)$ upper bound, depending on the number of polygons in the largest drawing. Hence, the matching can be achieved in polynomial time which tackles the computational barrier. We stand apart from the prior approaches by grouping low-level primitives into polygons and then considering their matching at this high-level point of view. Once polygons are mapped, it is interesting to take a closer look at a lower level by checking out segment layouts within the mapped polygons. This presents some advantages as elements are locally affected to define a local dissimilarity measure which is visually interesting; it makes easier the spotting of mis-detected areas. A complete data flow process for polygonized document evaluation is proposed. Our contribution in this domain is two-fold. Firstly, we compare a ground truth document and a computer generated document thanks to an optimal framework that proceeds with object mapping. Finally, another operator provides estimates on the relation between the segments within two mapped polygons in terms of edit operations, by means of a cycle graph matching. Figure 1 depicts an overview of our methodology.

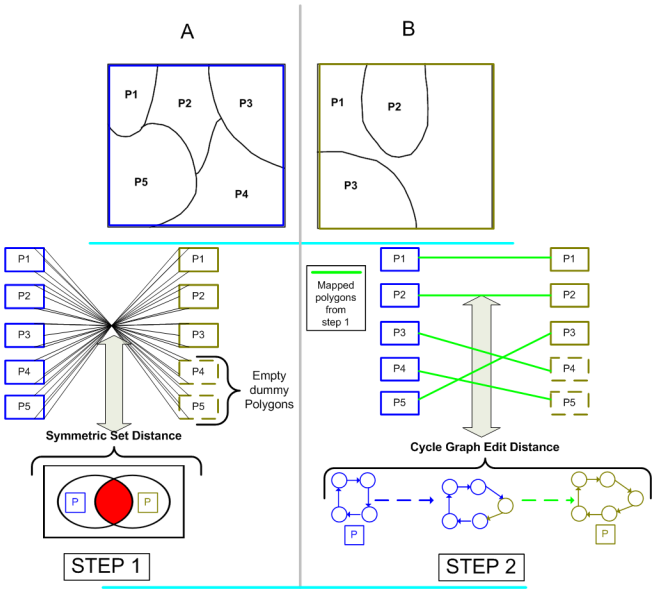


Fig. 1. Overview of the overall methodology. A bipartite graph weighted by the symmetric difference, and cycle graph edit distance applied to mapped polygons

1.4 Organization

The organization of the paper is as follows: Sect. 2.1 describes theoretically and in terms of algorithm the polygon mapping method, Sect. 2.2 explains the cycle graph matching process in order to assess the quality of the polygonal approximation. Sect. 2.3 sets forth the type of errors that are likely to occur in object retrieval systems. Sect. 3 describes the experimental protocol, this section also explains how to interpret our new set of indices in a application to cadastral map evaluation. A summary is included in Sect. 4, followed by discussions and concluding remarks.

2 A set of indices for polygonization evaluation

In this section, we define the two criteria involved in our proposal for a performance evaluation tool dedicated to polygonization. In the first part, a polygon assignment method is described. It aims at taking into account shape distortions caused by retrieval systems. Secondly, a matched edit distance is defined. This measure represents the variations introduced when a given system approximates digital curves. It is a synthesis on vectorization precision. Finally, mis-detection or over-detection

errors due to raster to polygon conversion are introduced in a third part. This breakdown leads to the definition of specific notations and error categorizations.

2.1 Polygon mapping using the Hungarian method

Once polygons are located within the vectorized document, it can be seen as a partition of polygons. Comparing two documents (D_1, D_2) comes down to matching each polygon of D_1 with each polygon of D_2. This assignment is performed using the Hungarian method which is formally described in the next part.

Algorithmic of the Hungarian method
Our approach for vectorized document comparison is based on the assignment problem. The assignment problem considers the task of finding an optimal assignment of the elements of a set D_1 to the elements of a set D_2. Without loss of generality, we assume that $|D_1| \geq |D_2|$. The complete bipartite graph $G_{pm} = D_1 \cup D_2 \cup \triangle, D_1 \times (D_2 \cup \triangle)$, where $\triangle$ represents empty dummy polygons, is called the polygon matching of D_1 and D_2. A polygon matching between D_1 and D_2 is defined as a maximal matching in G_{pm}. We define the matching distance between D_1 and D_2, denoted by $PMD(D_1, D_2)$, as the cost of the minimum-weight polygon matching between D_1 and D_2 with respect to the cost function K. The cost function is especially dedicated to our problem and is fully explained in section 2.1. This optimal polygon assignment induces a univalent vertex mapping between D_1 and D_2, such that the function $PMD : D_1 \times (D_2 \cup \triangle) \rightarrow \mathbb{R}_0^+$ minimizes the cost of polygon matching. If the numbers of polygons are not equal in both documents, then empty "dummy" polygons are added until equality $|D_1| = |D_2|$ is reached. The cost to match an empty "dummy" polygon is equal to the cost of inserting a whole unmapped polygon ($K(\emptyset, P)$). A shortcoming of the method is the one-to-one mapping aspect of the algorithm, however, this latter is performed at a high level of perception where data are less likely to be fragmented. Finally, this disadvantage should not discourage the use of the PMD distance considering the important speed-up it provides while being optimal, deterministic and quite accurate. In addition, unmapped elements are not left behind, they are considered either as "false alarm" or "false negative" according to the kind of mistakes they induced (see section 2.3).

Cost function for polygon assignments
Munkres' algorithm as introduced in the last section provides us an optimal solution to the assignment problem in $O(n^3)$ time. In its generic form, the assignment problem considers the task of finding an optimal assignment of the elements of a set GT to the elements of a set CG assuming that numerical costs are given for each assignment pair. In fact, a cost function does exist between each pair of polygons to express numerically their similarity, in the same way a "zero" will represent two identical polygons and "one" two polygons not sharing any common features. The polygon overlay, inspired by the theory of sets, measures the similarity between polygons. When polygons are compared into the same axis system, the overlay takes into account spatial adjustment between polygons. The process of overlaying polygons shares common points with set theory. Let's assume that A and B are two sets, the intersection can be reformulated through the set theory. *Intersection*, where the result includes all those set parts that occur in A and B. A way to compare them is to find out how A differs from B (see figure 2): In mathematics, the difference of two sets is the set of elements which are in one of the sets, but not in both. This operation is the set-theoretic kin of the exclusive disjunction in Boolean logic. The symmetric difference of the sets A and B is commonly denoted by $A \Delta B$. The symmetric difference is equivalent to the union of both relative complements, that is:

$$A \Delta B = (A \setminus B) \cup (B \setminus A)$$

and it can also be expressed as the union of the two sets, minus their intersection:

$$A \Delta B = (A \cup B) \setminus (B \cap A) \tag{1}$$

The symmetric difference is commutative and associative:

$$A \Delta B = B \Delta A$$

The empty set is neutral, and every set is its own inverse:

$$A \Delta \emptyset = A$$

$$A \Delta A = \emptyset$$

The set difference can be expressed as the union of the two sets, minus their intersection and represents the number of shared terms. In order to define measures of dissimilarity between complex objects (sets, polygons,...), a suitable possibility is to settle the measure on the quantity of shared terms. Based on this paradigm, the simplest similarity measure between two complex objects o_1 and o_2 is the matching coefficient mc :

$$mc = \frac{o_1 \wedge o_2}{o_1 \vee o_2} \tag{2}$$

Where $o_1 \wedge o_2$ denotes the intersection of o_1,o_2 and $o_1 \vee o_2$ stands for the union between the two objects.
Derived from the equation 2, dissimilarity measures which take into account the maximal common shared terms (mcs) of two sets were put forward:

$$d(o_1, o_2) = 1 - \frac{mcs(o_1, o_2)}{max(\mid o_1 \mid, \mid o_2 \mid)} \tag{3}$$

Where $\mid o \mid$ denotes the size of o. From the equation 2, the expression $o_1 \vee o_2$ is substituted by the size of the largest object and the intersection of two objects ($o_1 \wedge o_2$) is represented by the maximum common subset.

Finally, to obtain a dissimilarity measure between polygons, let us define a function K that induces a mapping of $P1$ and $P2$ into $\mathbb{R}$ ($K : P1 \times P2 \rightarrow \mathbb{R}$) where $P1$ and $P2$ are two polygons:

$$K(P1, P2) = 1 - \frac{mcs(P1, P2)}{\mid P1 \mid + \mid P2 \mid - mcs(P1, P2)} \quad (4)$$

Where $|P|$ denotes the area of the polygon P and where $mcs(P1, P2)$ is the largest set common to $P1$ and $P2$, i.e. it cannot be extended to another common set by the addition of any element. Straightforwardly, the largest common subset between $P1$ and $P2$ can be seen as the area in interception between $P1$ and $P2$. Consequently, equation 4 can be re-written as follows:

$$K(P1, P2) = 1 - \frac{|P1 \cap P2|}{|P1| + |P2| - |P1 \cap P2|} \quad (5)$$

In a general way, the equation 3 has been proved to be a metric in [26] and [27]. As long as, $|P1 \cap P2|$ is a valid maximum common shared terms for polygons, the metric properties hold true for the equation 5.

Now, let us take a closer look at the basic properties of this dissimilarity measure :

$$K(P1, P2) >= 0 \ \ \forall P1, P2 \quad (6)$$

$$K(P1, P2) = 0 \ \rightarrow \ P1 = P2 \quad (7)$$

$$K : [0; 1] \quad (8)$$

Theoretical discussion on our dissimilarity measure
Two questions are addressed in this part, the first one concerns the unity of PMD when facing heterogeneous documents (different scales and orientations) and the second point is devoted to the proof of PMD as being a metric. This point is crucial to demonstrate the ability of providing a rank information which is representative of the error level of a given document with respect to the entire collection.

The cost function behavior: One condition imposed by the Hungarian method is that the cost function has to be strictly positive or zero, this assumption is respected (see equation 6). In addition, the normalization between zero and one (see equation 5, equation 8) confers some interesting aspects to the distance.

Without normalization, the distance is highly dependent on the polygon surfaces. A higher importance would be given to large polygons and they could highly impact the final score, while small polygons would not be treated with significance.

Hence, this measure considers with equity whether the concerned polygons are small or not. It means that no bias will be introduced when facing large polygons.

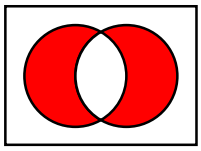


Fig. 2. $A \Delta B$

Thereby, a highly over-segmented vectorization with many small areas will be roughly as bad as an under-segmented vectorization with few large polygons. Finally, the normalization leads to a lower and an upper bound (equation 8) of our distance which is useful to compare a document collection with different scales. In this way, PMD is dependent on scale, translation and rotation variations. Nevertheless, these are desired properties for a distance which is meant to represent the exactness between two polygonized drawings.

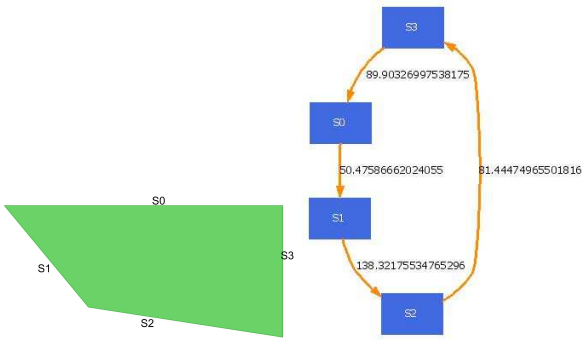
The Polygon Matching Distance is a metric (PMD):

Proof. To show that our measure of similarity between documents is a metric, we have to prove four properties for this similarity measure.

- $PMD(D_1, D_2) \geq 0$
  The polygon matching distance between two documents is the sum of the cost for each polygon matching. As the cost function is non-negative, any sum of cost values is also non-negative.
- $PMD(D_1, D_2) = PMD(D_2, D_1)$
  The minimum-weight maximal matching in a bipartite graph is symmetric, if the edges in the bipartite graph are undirected. This is equivalent to the cost function being symmetric. As the cost function is a metric, the cost for matching two polygons is symmetric. Therefore, the polygon matching distance is symmetric.
- $PMD(D_1, D_3) \leq PMD(D_1, D_2) + PMD(D_2, D_3)$
  As the cost function is a metric, the triangle inequality holds for each triple of documents in D_1, D_2 and D_3 and for those polygons that are mapped to an empty polygon. The polygon matching distance is the sum of the cost of the matching of individual polygons. Therefore, the triangle inequality also holds for the polygon matching distance.
- $PMD(D_1, D_2) = 0 \Rightarrow D_1 = D_2$
  If one of the polygons of D_1 cannot be matched exactly with a polygon of D_2 then $PMD(D_1, D_2) > 0$. A straightforward interpretation of this fact leads to the uniqueness property. Where all D_1' polygons are matched with a cost of zero to the polygons of M_2, it implies $D_1 = D_2$.

2.2 Matched edit distance for polygon comparison

The Hungarian method provides a formal framework to perform a one to one mapping between polygons. Each mapped pair of polygons minimizes its symmetric difference providing a topological information. However, this



(a) Polygon (b) Cycle Graph

Fig. 3. From polygon to cycle graph

measure does not take into account the labor work that has to be done to change a polygon from the CG to a correct polygon from the GT. In order to compensate this weakness, we decide to include an additional measure which reveals how many edit operations have to be done to change a polygon into another according to some basic operations. That's why we present an edit distance for polygon comparison. From definition 2 and figure 3, a clear link exists between a polygon and its representation by a cycle graph. The next part defines a Cycle Graph Edit Distance (CGED) for polygon comparison, this latter deals with the graph matching problem applied cycle graph.

A Cycle Graph Matching Distance for Polygon Comparison

Visually, two chains of segments are similar if the length attributes and angles between consecutive segments can be aligned. In the literature on polygonal shape recognition, most approaches base the distance definition between two polygonal shapes on length and angle differences. For example, Arkin et al. used in [28] the turning function which gives the angle between the counterclockwise tangent and the x-axis as a function of the arc length. Their results are in accordance with the intuitive notion of shape similarity. More recently, in [29], Lladós et al. represented regions by polylines and string matching techniques are used to measure their similarity. The algorithm follows a branch and bound approach driven by the RAG edit operations. This formulation allows matching computing under distorted inputs. The algorithm has been used for recognizing symbols in hand drawn diagrams.

Polygonal shapes require to characterize the segments (their length) but also their relationships by the angle information.

The graph-based representation was preferred to string representation. In fact, the protocol is designed for polygons but may also be extended to other line shapes, for instance this could be made by completing the graph representation to connected vectors instead of searching for cyclic polygons. In this way, the graph-based viewpoint could be the container of a wider range of entities. It leaves the door open to a more global paradigm, the object matching question.

The concept of edit distance has been extended from strings to trees and to graphs [30], [31]. Similarly to string edit distance, the key idea of graph edit distance is to define the dissimilarity, or distance, of graphs by the minimum amount of distortion that is needed to transform one graph into another. Compared to other approaches to graph matching, graph edit distance is known to be very flexible since it can handle arbitrary graphs and any type of node and edge labels. Furthermore, by defining costs for edit operations, the concept of edit distance can be tailored to specific applications. A standard set of distortion operations is given by insertions, deletions, and substitutions of both nodes and edges. We denote the substitution of two nodes u and v by $(u \rightarrow v)$, the deletion of node u by$(u \rightarrow \lambda)$, and the insertion of node v by $(\lambda \rightarrow v)$. For edges we use a similar notation.

Given two graphs, the source graph G_1 and the target graph G_2, the idea of graph edit distance is to delete some nodes and edges from G_1, relabel (substitute) some of the remaining nodes and edges, and insert some nodes and edges in G_2, such that G_1 is finally transformed into G_2.

A sequence of edit operations $e_1; ...; e_k$ that transforms G_1 completely into G_2 is called an edit path between $G_1$and G_2. Obviously, for every pair of graphs $(G_1; G_2)$, there exist a number of different edit paths transforming G_1 into G_2. To find the most suitable edit path, one introduces a cost for each edit operation, measuring the strength of the corresponding operation. The idea of such a cost function is to define whether or not an edit operation represents a strong modification of the graph. Obviously, the cost function is defined with respect to the underlying node or edge labels. Clearly, between two similar graphs, there should exist an inexpensive edit path, representing low cost operations, while for dissimilar graphs an edit path with high costs is needed. Consequently, the edit distance of two graphs is defined by the minimum cost edit path between two graphs. The computation of the edit distance is carried out by means of a tree search algorithm which explores the space of all possible mappings of the nodes and edges of the first graph to the nodes and edges of the second graph.

Definition 3. *(Cycle Graph Matching)*

In this work, the problem which is considered concerns the matching of cycle directed labeled graphs. Such graphs can be defined as follows: Let L_V and L_E denote the set of node and edge labels, respectively. A labeled graph G is a 4-tuple $G = (V, E, \mu, \xi)$, where

- *V is the set of nodes,*
- *$E \subseteq V \times V$ is the set of edges*
- *$\mu : V \rightarrow L_V$ is a function assigning labels to the nodes, and*
- *$\xi : E \rightarrow L_E$ is a function assigning labels to the edges.*
- *$|V|=|E|$*

Now, let us define a function CGED based on the Cycle Graph Edit Distance that induces a mapping of

Table 1. Edit costs

	Node	Edge
Label Substitution	$\gamma((l_i^A) \rightarrow (l_j^B)) = \left\lvert \frac{l_i^A}{\lvert A\rvert} - \frac{l_j^B}{\lvert B\rvert} \right\rvert$	$\gamma((\Phi_i^A) \rightarrow (\Phi_j^B)) = \frac{\lvert \Phi_i^A - \Phi_j^B \rvert}{360}$
Addition	$\gamma(\lambda \rightarrow (l_j^B)) = \frac{l_j^B}{\lvert B\rvert}$	$\gamma(\lambda \rightarrow (\Phi_j^B)) = \frac{\lvert \Phi_j^B \rvert}{360}$
Deletion	$\gamma((l_i^A) \rightarrow \lambda) = \frac{l_i^A}{\lvert A\rvert}$	$\gamma((\Phi_i^A) \rightarrow \lambda) = \frac{\lvert \Phi_i^A \rvert}{360}$

G_1 and G_2 into $\mathbb{R}$ ($CGED : G_1 \times G_2 \mapsto \mathbb{R}$) :

$$CGED(G_1, G_2) = \min_{(e_1,..,e_k)\in\gamma(G_1,G_2)} \sum_{i=1}^{k} (edit(e_i))$$

Where $\gamma(G_1, G_2)$ denotes the set of edit paths transforming G_1 into G_2, and edit denotes the cost function measuring the strength $edit(e_i)$ of edit operation e_i
.

A Cycle Graph is a cycle which visits each vertex exactly once and also returns to the starting vertex. As a consequence, the set of all edit paths is considerably reduced and the Cycle Graph matching can be solved in $O(nm \log m)$ time.

In order to use cycle graph matching for polygon accuracy evaluation, we use an attributed graph representation. Starting from a polygonal approximation of the shape, a graph is built. We use the segments as primitives, encoding them with a set of nodes. Each node is labeled with a real number l_i, where l_i denotes the length of the segment s_i. Then, edges are built using the following rule: two nodes are linked with a directed and attributed edge if two constitutive segments share a common point. Each edge is labeled with a real number Φ_i that denotes the angle between s_i and s_{i-1} in the counterclockwise direction. We can appreciate an example of how these attributes are computed for a sample shape in figure 3.

Let A and B be two chains of adjacent segments, represented as cycle graphs, with total lengths $|A|$ = n and $|B|$ = m and with respectively attributed cycle graph representations:

$$G_A = (V^A, E^A) = (l_i^A ... l_n^A), (\Phi_i^A ... \Phi_n^A)$$

and,

$$G_B = (V^B, E^B) = (l_i^B ... l_m^B), (\Phi_i^B ... \Phi_m^B)$$

The cost functions for attributed cycle graph matching are reported in table 1

Interpretation of the edit operations
There are the proposed cost functions inspired by the ones designed by Tsay and Tsai in [32] where they use string matching for shape recognition.

The operation attributes decrease the edit costs for primitives undergoing noisy transformations, as the inherent segment fragmentation from the raster-to-vector process. And it aims to compare polygons with different number of segments making the system tolerant to segment cardinality.

Furthermore, our edition functions can describe real transformations applied to polygons. When editing a vectorization, basic operations are: Remove, Add or Move a segment; a visual illustration of theses operations is given in figure 4. Through the linear combination of the cost functions, it is possible to recreate the usages of a person modifying a vectorization, and the definitions below present the combinations to obtain different polygon transformations. Conceptually, we are here very close to the ideas proposed by Chhabra [5].

Reasoning in a segment deletion context, we proceed by two simple case analysis to detail the elaboration of the generated operation sequence and then to describe the impact of the involved cost functions.

Case 1. Edit operation for segment deletion
At first, deleting a segment comes with the deletions of a node and an edge into the graph representation.
We conclude that

$$deletions = \gamma((l_i^A) \rightarrow \lambda) + \gamma((\Phi_i^A) \rightarrow \lambda)$$

In addition, the deletions of a node and an edge in a cycle graph creates an orphan edge that must be reconnected. To take into account this modification, an edge label substitution is operated.
Consequently we conclude:

$$substitution = \gamma((\Phi_i^A) \rightarrow (\Phi_j^B))$$

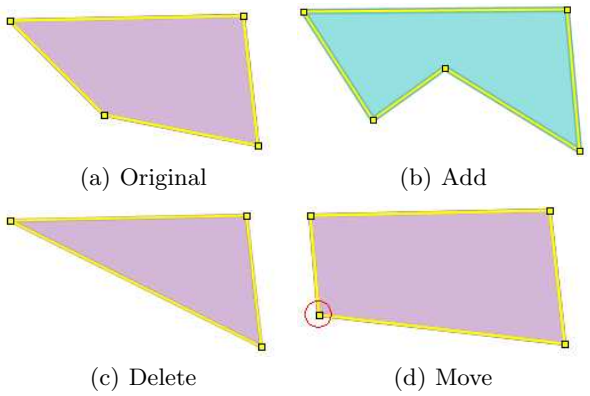
Finally, the sequence of operations is:

$$\gamma(s_i \rightarrow \lambda) = deletions + substitution$$

Plainly, we change the formal expressions by their corresponding costs to obtain the following formula:

$$\gamma(s_i \rightarrow \lambda) = \frac{l_i^A}{|A|} + \frac{\Phi_i^A}{360} + \frac{|\Phi_i^A - \Phi_j^B|}{360}$$

Case 2. Cost function for a segment deletion
A segment deletion is operated to remove a segment erroneously inserted by a given R2V system. We can show that the cost function associated to the edit operation $\gamma(s_i \rightarrow \lambda)$ is representative of the mistake importance. We proceed by a simple case analysis. The global cost function is impacted by two parameters, the size and the misalignment of the removed segment. $\gamma(s_i \rightarrow \lambda)$ expresses two deformations: (i) How large was the segment to be removed and (ii) How misaligned was the segment to be removed compared with the adjacency vectors. In this way, larger is the segment to be removed and bigger is the mistake made by the R2V. The cost function responds correctly, larger is the segment and bigger is the quantity $\gamma(s_i \rightarrow \lambda)$ by the increase of the $\frac{l_i^A}{|A|}$ factor. The second kind of errors generated by R2V systems is the segment misalignment. Larger is the angle variations of the removed segment with the adjacency segments in the polygon and bigger is the committed error. Accordingly,



(a) Original (b) Add

(c) Delete (d) Move

Fig. 4. Basic edit operations applied to a polygon.

larger is the angle variations of the removed segment and bigger is the quantity $\gamma(s_i \rightarrow \lambda)$ by the increase of $\frac{\Phi_i^A}{360} + \frac{|\Phi_i^A - \Phi_j^B|}{360}$ combination.

Thereby, we provide the definitions for segment deletion, addition and move.

Definition 4. *Segment deletion transformation*

$$\gamma(s_i \rightarrow \lambda) = \gamma((l_i^A) \rightarrow \lambda) + \gamma((\Phi_i^A) \rightarrow \lambda) + \gamma((\Phi_i^A) \rightarrow (\Phi_j^B))$$

$$\gamma(s_i \rightarrow \lambda) = \frac{l_i^A}{|A|} + \frac{\Phi_i^A}{360} + \frac{|\Phi_i^A - \Phi_j^B|}{360}$$

Definition 5. *Segment addition transformation*

$$\gamma(\lambda \rightarrow s_j) = \gamma(\lambda \rightarrow (l_j^A)) + \gamma(\lambda \rightarrow (\Phi_j^A)) + \gamma((\Phi_i^A) \rightarrow (\Phi_j^B))$$

$$\gamma(\lambda \rightarrow s_j) = \frac{l_j^A}{|A|} + \frac{\Phi_j^A}{360} + \frac{|\Phi_i^A - \Phi_j^B|}{360}$$

Definition 6. *Segment move transformation*

$$\gamma(s_i) \rightarrow (s_j) = \gamma((l_i^A) \rightarrow (l_j^B)) + \gamma((\Phi_i^A) \rightarrow (\Phi_j^B))$$

$$\gamma(s_i) \rightarrow (s_j) = \left| \frac{l_i^A}{|A|} - \frac{l_j^B}{|B|} \right| + \frac{|\Phi_i^A - \Phi_j^B|}{360}$$

Matched Edit Distance

To complete the process, the Cycle Graph Matching Distance ($CGED$) has to be performed on every pair of mapped polygons found by the Hungarian methods when it is based on the symmetric difference. Note that if one polygon is associated to an empty dummy item then the cycle graph matching distance will be only composed of addition operations. The Matched Edit Distance(MED) is then composed of the sum of all $CGED(G_1, G_2)$ computed on every pair of graphs extracted from the polygons.

$$MED(D_1, D_2) = \frac{\sum_{i=1}^{\max(|(D_1|,|D_2|)} CGED(G_i^{D_1}, G_i^{D_2})}{\max(|(D_1|, |D_2|)}$$

2.3 Types of error and notations

Here, we sum up a set of two criteria which will help us to evaluate a given raster to vector conversion. Each measure is a viewpoint on the vectorization process. However, every criterion can still be divided into two categories according to the nature of the error it expresses. Hence, the next part defines the different kinds of errors that can occur when dealing with object retrieval systems.

Types of error

- Type I error, also known as an "error of the first kind", a false alarm or a "false positive": the error of rejecting a null hypothesis when it is actually true. Plainly speaking, it occurs when we are detecting a polygon when in truth there is none, thus indicating a test of poor specificity. An example of this would be if an application should retrieve a polygon when in reality there is none. Type I error can be viewed as the error of excessive credulity.
- Type II error, also known as an "error of the second kind", a "false negative": the error of failing to reject a null hypothesis when it is in fact not true. In other words, this is the error of failing to detect a polygon when in truth there is one, thus indicating a test of poor sensitivity. An example of this would be if a test should show that there is no polygon when in reality there is one. Type II error can be viewed as the error of excessive skepticism.

Notations

For the understanding of these tests, we first introduce notations that will make the reading much simpler. A dissimilarity measure between vectorized documents is a function :

$$d : X \times X \rightarrow \mathbb{R}$$

where X is a vectorized document. We report in table 2, the notations derived from this general form.

X_{tp} puts forward the cost involved when matching a pair of mapped polygons. This is a synonym of accuracy, it denotes how well suited is the detected polygon from the D_{CG}. X_{fp} takes the stock of the over-detections issued from the raster to vector conversion step. On the other hand, X_{fn} represents the mis-detections, it occurs when the software used to vectorize has a strict policy of rejection which leads to an under-detection of objects. For clarity reasons, when no precision is specified, X refers to X_{all}. Finally, a desirable information is the number of false alarms, false negative and true positive polygons retrieved by the retro-conversion system. These values are normalized as follows to obtain a comparable rate between documents.

$$\eta_{fn} = \frac{\sharp \text{ of mis-detected polygons}}{\max(|(D_1|, |D_2|)}$$

$$\eta_{fp} = \frac{\sharp \text{ of over-detected polygons}}{\max(|(D_1|, |D_2|)}$$

Notation	Method	Type of error	Distance
PMD_{tp}	PMD	True Positive	symmetric difference
PMD_{fn}	PMD	False Negative	symmetric difference
PMD_{fp}	PMD	False Positive	symmetric difference
PMD_{md}	PMD	Mis-detection (fn+fp)	symmetric difference
PMD_{all}	PMD	(tp+fn+fp)	symmetric difference
MED_{tp}	MED	True Positive	Cycle Graph Matching
MED_{fn}	MED	False Negative	Cycle Graph Matching
MED_{fp}	MED	False Positive	Cycle Graph Matching
MED_{md}	MED	Mis-Detection (fn+fp)	Cycle Graph Matching
MED_{all}	MED	(tp+fn+fp)	Cycle Graph Matching
η_{tp}	-	True Positive	♯ of well-detected polygons
η_{fp}	-	False Positive	♯ of over-detected polygons
η_{fn}	-	False Negative	♯ of under-detected polygons

Table 2. Distance between vectorized documents.

3 Experiments

This section is devoted to the experimental evaluation of the proposed approach. Firstly, we describe databases that are used to benchmark our measures. Then the protocol of our experiments is defined by enumerating the kind of assessments we performed. The two first tests are dedicated to graphical symbols from GREC contests. On this basis, we aim at illustrating the ability of Polygon Matching Distance (PMD) and Matched Edit Distance (MED) of being representative of polygon deformations (shape variation and polygonal approximation modification, respectively). The last evaluation concerns the cadastral map subject, we show results on a large collection of maps. We provide guidelines to understand the meaning of our set of indices. In this pedagogic objective, a visualization of detection errors is proposed.

In this practical work, methods were implemented in Java 1.5 and run on a 2.14GHz computer with 2G RAM. Both databases and performance evaluation tools are freely available on this web site:

http://alpage-l3i.univ-lr.fr/

Both datasets and the experimental protocol are firstly described before investigating and discussing the merits of the proposed approach.

Table 3. Characteristics of the cadastral map collection: Base C

	♯ of polygons	♯ of vectors
$\|GT\|$	2335	654017
$\|CG\|$	2626	850667
mean $\|GT\|$	23.35	64.75
mean $\|CG\|$	26.26	85.06
max $\|GT\|$	83	101
max $\|CG\|$	69	100

Table 4. Characteristics of the symbols data sets: Base A, B

	Base A	Base B
Number of classes (N)	70	53
$\mid Base \mid$	350	371
Noise type	Vectorial	Binary
Noise levels	4	6
Assessment purpose	Shape distortion	Digital curve approximation

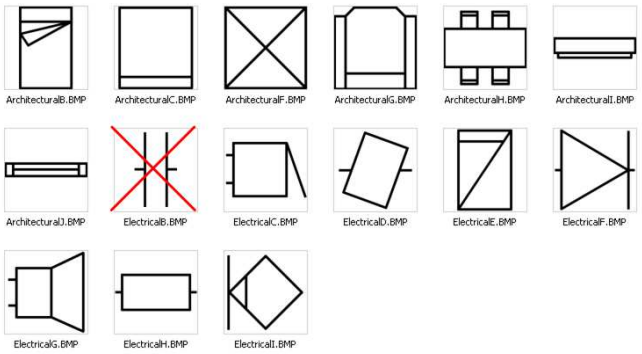


Fig. 5. A sample among the seventy symbols used in our ranking test. Polygon-less symbols were removed.

3.1 Databases in use

In recent years the subject of performance evaluation has gained popularity in pattern recognition and machine learning. In the graphics recognition community, a huge amount of efforts was made to elaborate standard and publicly available data sets. Especially, E. Valveny [33],[34] and M. Delalandre [35] published on-line symbol datasets for a symbol classification purpose. In this section, we describe two databases derived from [34] and [35] and we also present a cadastral map collection. The content of each database is summarized in tables 3, 4.

Base A: Shape distortion. The paper presented in [35] gave birth to a publicly available database of symbols[1]. From this setting, we removed all polygon-less symbols to fit our purpose which was to evaluate polygon detection methods. Hence, we selected 70 symbols from the GREC'05 contest [36] and a sample is presented in figure 5.

[1] *http : / / mathieu. delalandre.free.fr /projects /sesyd /sketches.html*

On perfect symbols, a vectorial noise is applied to generate a collection of degraded elements. We could not afford to use real data because of the difficulty of collecting images with all kinds of transformations and noise. Besides, it is not easy to quantify the degree of noise in a real image. Then, it is not possible to define a ranking of difficulty of images according to the degree of noise. In our experiments, we have re-used methods for the generation of shape transformation (based on active shape models [33]).

Vectorial Distortion: The goal of vectorial distortion is to deform the ideal shape of a symbol in order to simulate the shape variability produced by hand-drawing. The method for the generation of vectorial distortions of a symbol is based on the Active Shape Models [37]. This model aims to build a model of the shape, statistically capturing the variability of a set of annotated training samples. In order to be able to apply this method, we need to generate a good set of training samples. This is not a straightforward task due to the statistical nature of the method. The number of samples must be high enough, and the samples must reflect the usual kind of variations produced by hand-drawing. However, it is difficult to have a great number of hand-drawn samples of each symbol. To be really significant, these samples should be drawn by many different people. Thus, the decision of generating automatically the set of samples has arisen. Based on the generation of deformed samples through the random modification of a different number of vertices of the symbol each time [38].

Each sample is represented using the model described in [39], which permits easy generation of deformed shapes. Each symbol is described as a set of straight lines, and each line is defined by four parameters: coordinates of mid-point, orientation and length. Thus, each deformed sample can be seen as a point x_i in a $4n$ dimensional space, where n is the number of lines of the symbol. Then, principal component analysis (PCA) can be used to capture the variability in the sample set. Given a set of samples of a symbol, we can compute the mean $\overline{x}$ and the covariance matrix S. The main modes of variation are described by the first eigenvectors p_k of the covariance matrix S. The variance explained by each eigenvector is equal to its corresponding eigenvalue. Thus, each shape in the training set can be approximated using the mean shape and a weighted sum of the eigenvectors:

$$x = \overline{x} + Pb$$

where $P = p_1, ..., p_m$ is the matrix of the first m eigenvectors and b is a vector of weights. This way, new images of a symbol can be generated by randomly selecting a vector of weights b. Increasing values of b_i will result in increasing levels of distortion (see figure 6).

The model of vectorial distortion described in the former paragraph has been applied with four increasing levels of distortion to generate 280 (70*4) images of symbols. The variance was tuned from 0.00025 to 0.00100 by step of 0.00025. This way of changing the variance is coherent with the protocol presented in [35]. The entire database is then made up of 350 elements, 280 degraded symbols and 70 models. The shape distortion generator, 3gT system was provided by M. Delalandre[2]. 3gT "generation of graphical ground Truth" is a system to generate random graphical documents (exported into SVG) of low-level primitives (vectors, circles, ...) with their corresponding ground truth. Base A is a reliable source to evaluate the shape distortion sensitivity of our polygon location measure. Numerical details concerning this data set are presented in table 4.

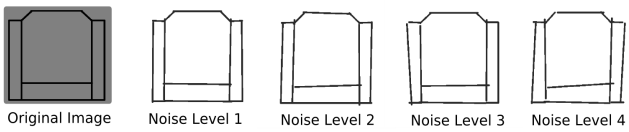


Fig. 6. Examples of increasing levels of vectorial distortion

Base B: Binary degradation. First of all, we decided to use the data set provided by the GREC'03 contest. Mainly two application domains, architecture and electronics, were adopted as a representative sample of a wide range of shapes. GREC'03 database is originally constituted of 59 symbols from which we removed symbols without polygons. This pruning step led us to a database of 53 symbols. From the 9 noise levels available, we only focused on the 6 first levels. Consequently, database B is made up of 318 (6*53) binary degraded symbols plus 53 ideal models, i.e. a total of 371 polygonized elements according to the process explained in the next paragraph.

Binary Degradation: Kanungo et al. have proposed a method to introduce some noise on bitmap images [40]. The purpose of this method is to modelize noises obtained by operations like printing, photocopying, or scanning processes. The problem is approached from a statistical point of view. The core principle of this method is to flip black and white pixels by considering, for each candidate pixel, the distance between it and the closest inverse region. The degradation method is validated using a statistical methodology. Its flexibility in the choice of the parameters requires some adaptations. Indeed, a large set of degradations can be obtained. The method itself accepts no less than 6 parameters, allowing to tune the strength of white and black noise, the size of the influence area of these noises, a global noise (which does not depend of the presence of white/black pixels), and a post-processing closing based on well-known morphological operators. Of course, these 6 parameters may generate a large number of combinations, and thus, of models of degradation. So, if the core method used for the degradation tests is formal and validated for its correctness, the determination of the set of parameters used for the contest is more empirical. This framework was applied to the organization of the GREC'03 contest on symbol recognition. In [34], authors attempted to reproduce a set of degradations representing some realistic artifacts

[2] http://mathieu.delalandre.free.fr/projects/3gT.html

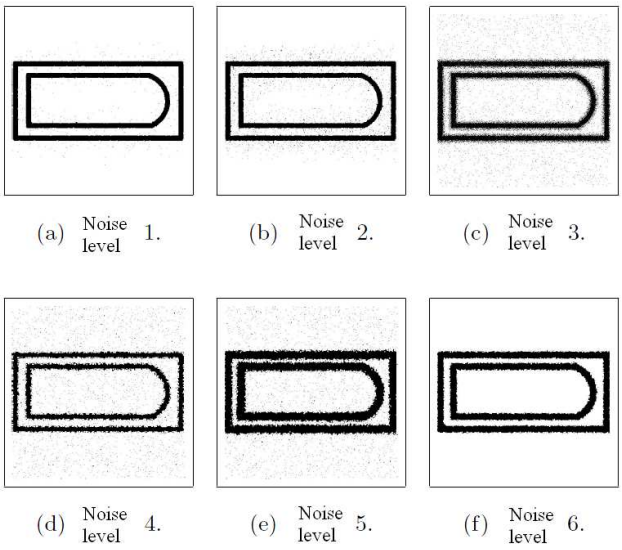


Fig. 7. Samples of some degraded images generated using the Kanungo method for each level of degradation used.

(to simulate noise produced when printing, photocopying and scanning). Six levels of degradation (see figure 7) were determined by [34]. They took care to represent some standard noises: local, global troubles.

Binary degradation impacts on polygonal approximations: The higher is the noise level the higher are the distortions on the polygonal approximation. The noise level has a direct influence on the vectorization algorithm. In this experiment, we used a standard data flow process to polygonize the symbols: (i) Cleaning[3]; (ii) Skeletonization; (iii) Polygonal approximation and (iv) Polygonizer. Arbitrarily, we adopted the well-known di Baja's skeletonizer [41] and the Wall and Danielsson's vectorization [42]. Then a polygonizer was applied to transform the set of segments into polygons. These steps are summed up in figure 8. A piece of polygon is zoomed-in to show the perturbation applied on the polygonal approximation when noise increases. The method only requires a single threshold i.e., the ratio between the algebraic surface and the length of the segments which makes this linear time algorithm fast and efficient. This parameter is set to 60 pixels for all the experiments. We did not want to assess the impact of the approximation threshold but rather the impact of noise on polygonization when the threshold is frozen.

More information concerning those data is detailed in table 4.

Base C: Cadastral map collection. In the context of a project called ALPAGE, a closer look is given to ancient French cadastral maps related to the Parisian urban space during the 19th century (figure 9). Hence, the map collection is made up of 1100 images coming from the digitalization of Atlas books. On each map, domain-objects called Parcels are drawn by using color

[3] A simple 5x5 median filter

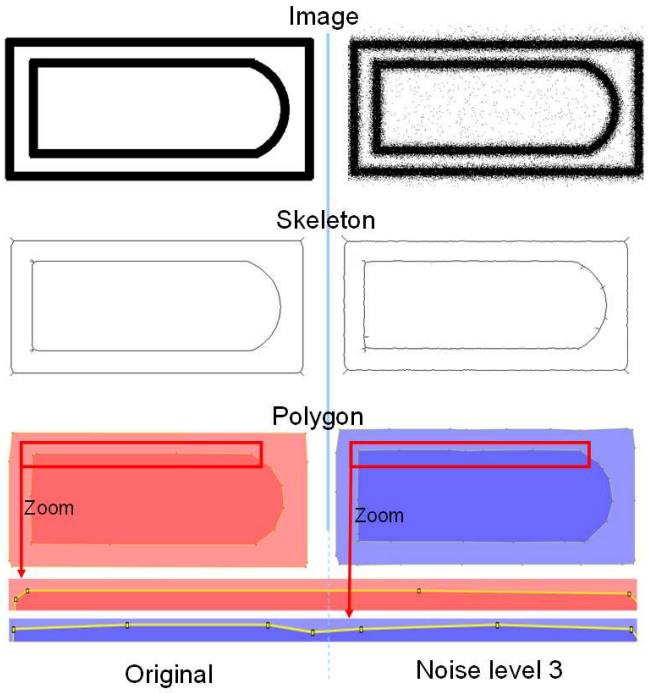


Fig. 8. Example of the polygonal approximation when increasing the noise level.

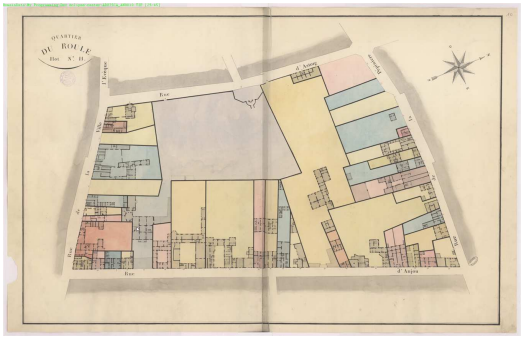


Fig. 9. Example of cadastral map (7616 x 4895 Pixels, 200 dpi, 24 BitsPerPixel)

to distinguish between them. From a computer science perspective, the challenge consists in the extraction of information from color documents with the aim of providing a vector layer to be inserted in a GIS (Geographical Information System).

Automatic polygon detection: In this project, a bottom-up strategy is adopted. In bottom-up strategies, algorithms are performed in a fixed sequence, usually starting low-level analysis of the gray level or black and white image, from which primitives are extracted. From this starting point, the four stages for extracting parcels from a cadastral map are put forward. (i) At first, a color gradient is performed to locate objects within the image. (ii) Then, a text/graphic segmentation is run on the gradient image to preserve only graphic elements [43], [44]. (iii) Thirdly, a digital curve approximation is performed to transform pixels into vectors [45]. (iv) finally, vectors

are gathered to form polgyons using a polygonizer algorithm [46]. The parcel extractor is evaluated using our set of indices.

Ground-Truthing: With the help of experts in several fields of sciences such as Historians, Archaeologists and Geographers, a campaign of handmade vectorization was carried out. This work was intensively labor consuming yet necessary. It was the only way to give us the opportunity to fully evaluate the accuracy of our work. The main goal was to build a reference database to investigate the merit of our parcel retrieval scheme. Manually, 100 raster maps were carefully and precisely vectorized to constitute a reliable collection of 2335 parcels of lands. These units are encoded as polygons according to the definition 2. Thereby, a real link does exist between a parcel and its polygon representation. This labor intensive procedure represents a real reference to measure up the accuracy and the validity of our automatic vectorization [44], [47]. The content of the database is summarized in table 3. In average, there are 25 parcels per map and this accounts for about 75 line segments per parcel. The ground-truth was manually made according to simple rules. Each parcel had to be described by a polygon. The median line was favored in the line tracking phase. The precision question was solved by imposing to fit at best the parcel contour and consequently, in each polygon, a vertex corresponds to a significant direction change. For each image of document, there exists exactly one pair of vectorized maps, one map called Ground Truth and one map named Computer Generated, respectively $< D_{GT}; D_{CG} >$. An example of a pair of vectorization to be matched is displayed in figure 10. Further details on this data set are presented in table 3. Note that this database is made up of real data.

A synthesis about this database is reported in table 3 while the content is publicly available at

http://alpage-l3i.univ-lr.fr/PE/alpagedb.zip.

3.2 Protocol

Three different ways of evaluating our indices are proposed.

Polygon Matching Distance evaluation: To assess the ability of the polygon mapping distance to increase when documents get badly reconstituted, we focus on Base A. Base A is representative of different shape distortions and consequently, polygon shapes are affected by this noise. On Base A, we performed a ranking test using PMD as a dissimilarity measure. A visual explanation of how ranks are obtained is brought to view in figure 11. Then, ranks are compared thanks to a statistical method called a Kendall's test and defined as follows:

Definition 7. *Kendall's test*

We assess the correlation concerning the responses to k-NN queries when using PMD as dissimilarity measures. The setting is the following: in a given Base X, we select a number N of symbols, that are used to query by similarity the rest of the dataset. Top k responses to each query obtained using PMD are compared with the ground-truth. The ground-truth ranks are obtained thanks to the control of noise level. The similarity of the PMD ranks and the ground-truth ranks is measured using Kendall correlation coefficient. We consider a null hypothesis of independence(H0) between the two responses and then, we compute, by means of a two-sided statistical hypothesis test, the probability (p-value) of getting a value of the statistic as extreme or more extreme than observed by chance alone, if H0 is true. The Kendall's rank correlation measures the strength of monotonic association between the vectors x and y (x and y may represent ranks or ordered categorical variables). Kendall's rank correla-

(a) GT

(b) CG

Fig. 10. Two vectorizations to be mapped ($|D_{CG}| = 46$ $|D_{GT}| = 40$).

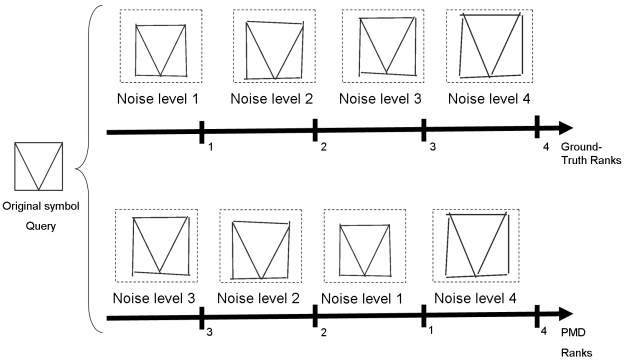


Fig. 11. Ranking explanation. Ranks 3 and 1 were swapped by PMD

tion coefficient τ may be expressed as

$$\tau = \frac{S}{D}$$

Where,

$$S = \sum_{i<j} (sign(x[j] - y[i]).sign(y[i] - x[j])) \quad (9)$$

And,

$$D = \frac{k(k-1)}{2} \quad (10)$$

Polygonal approximation sensitivity: In this second experiment, we aim at assessing the capacity of the Matched Edit Distance (MED) to increase when the polygonal approximation gets badly reconstituted by the retrieval systems. Base B is involved in this test. Base B is a binary degraded set of symbols and the higher is the noise level on symbols and the more disturbed is the polygonal approximation from the original one. In this way, we control the distortion level of the digital curve approximation and consequently, we obtain a ground-truth order from an ideal symbol by controlling the noise level, in figure 8. Finally, the ranks returned by MED and the ground-truth are compared according to the Kendall's test described in Def.7. using a Kendall test.

Application to the evaluation of parcel detection: Our last experiments are based on real data that composed Base C. At first, we performed the PMD distance on a single pair of given maps and this in order to highlight dissimilarities issued from the raster to vector conversion. Then, an experiment was dedicated to the evaluation of the entire collection of cadastral maps. We provided an interpretation of the results through the viewpoints of our set of indices. Finally, a statistical framework was described to figure out relations between the different indices.

3.3 Polygon Matching Distance evaluation

Using $N = 70$, $k = 4$ equal to the number of noise levels available in Base A, we present in figure 12 and table 5, the results obtained in terms of τ values. From the 70 tests, only 9 have a p-value greater than 0.05, so we can say that the hypothesis $H0$ of independence can be rejected in 87.4% cases, with a risk of 5%. The observed correlation between the responses to k-NN queries when using the ground-truth and Polygon Matching Distance (PMD) tends to reveal a rank relation between both (median value of $\tau = 0.800$). By stress testing a given system, we aim at demonstrating that our protocol can reveal strengths and weaknesses of a system. The PMD index increases when image degradation increases.

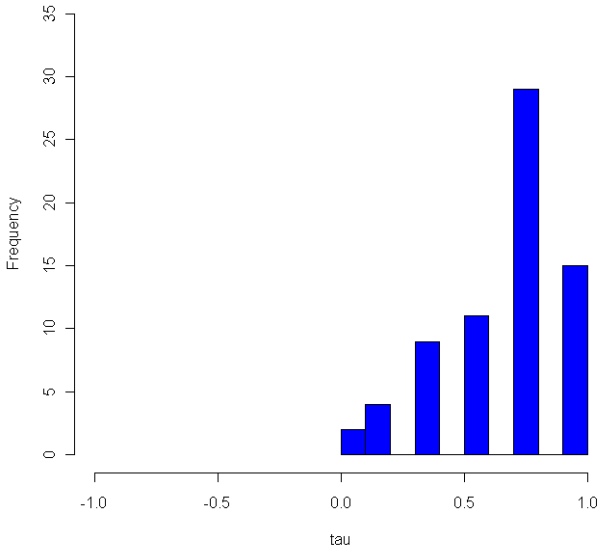


Fig. 12. Base A: Kendal correlation. Histogram of τ values obtained comparing ground-truth and PMD ranks.

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
τ	0.0000	0.6000	0.8000	0.7029	0.8000	1.0000

Table 5. Summary of Kendall correlation (τ). PMD *vs* ground-truth

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
τ	0.3333	0.6190	0.7143	0.7107	0.8095	1.0000

Table 6. Summary of Kendall correlation (τ). MED *vs* ground-truth

3.4 Polygonal approximation sensitivity

Using $N = 53$, $k = 6$ equal to the number of noise levels in Base A, we present in figure 13 and table 6, the results obtained in terms of τ values. From these results, we reject the null hypothesis of mutual independence between MED and the ground-truth rankings for the students. With a two-sided test we are considering the possibility of concordance or discordance (akin to positive or negative correlation). A one-sided test would have been restricted to either discordance or concordance, which would be an unusual assumption. In our experiment, we can conclude that there is a statistically significant lack of independence between MED and the ground-truth rankings of the symbols by MED. MED tended to rank symbols with apparently greater noise as being farther from the ideal symbol than those with apparently less noise and vice versa.

3.5 Application to the evaluation of parcel detection

A visual dissimilarity measure of local anomalies: In this part, we focus on comparing maps two by two. This

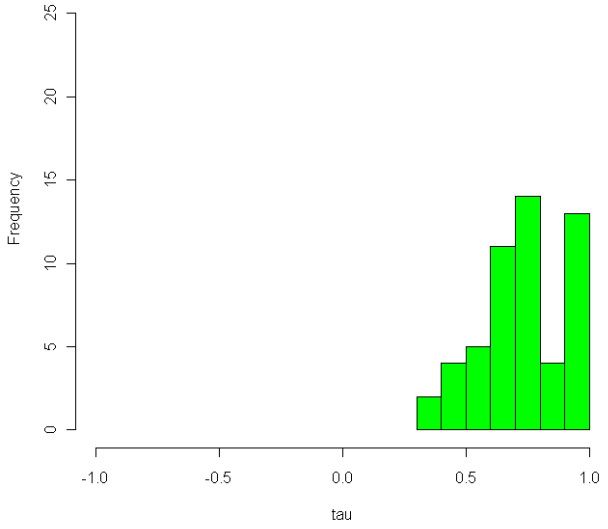


Fig. 13. Base B: Kendal correlation. Histogram of τ values obtained comparing ground-truth and MED ranks.

	X_{all}	X_{tp}	X_{fp}
PMD	0.1856	0.0552	0.1304
MED	0.5068	0.3764	0.1304
η	1	0.8695	0.1304

Table 7. Measures of performance.

difficult task requires a good observation of the local differences between the compared documents. On a randomly picked pair $< D_{GT}; D_{CG} >$, we computed the Polygon Matching Distance (PMD_{all}). A bi-dimensional representation of the costs to assign each element from the D_{CG} to the D_{GT} is displayed in figure 14, whereas values of the different measures are reported in table 7. Figure 14 provides a visual understanding of where the anomalies are located. Firstly, it facilitates the spotting of errors and other aberrations and especially, this framework can help domain experts understanding the limits and advantages of a vectorization software. Figure 14 is worth a thousand words, it makes easier the communication and the implementation of mutualized working tools for both Information and Communication Technologies (ICT) - Humanities and Social Sciences (HSS) communities.

To conclude, it can help users to spot where the mistakes are located and so save them a lot of time (time saver). It can help software designers to locate easily where the R2V conversion failed and consequently, this local visualization at a polygon level facilitates the categorization of detection errors.

Evaluation of a collection of maps: From the data set of vectorized maps, we attempted to evaluate the quality of the overall conversion process through the viewpoints

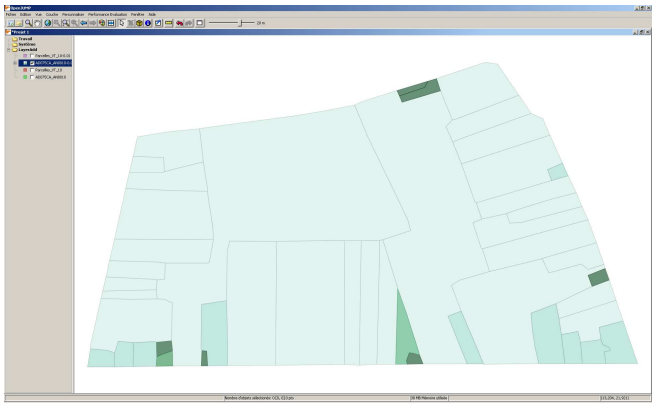


Fig. 14. Local dissimilarities between the two maps. The lighter the better. It means the darker is a parcel, the worst is its assignment. The overall cost=0.1856 can be broken down into a mis-detection cost, PMD_{fp}=0.1304 and a true positive cost PMD_{tp}=0.0552

offered by the two main criteria that we have described, PMD, MED.

Over the map collection, we observed in figure 15 an over-detection tendency. In average, 31% of the retrieved polygons are misleading. 71% of these wrong polygons are concerned by an over-detection behavior ($\overline{\eta_{fp}} = 0.22$).

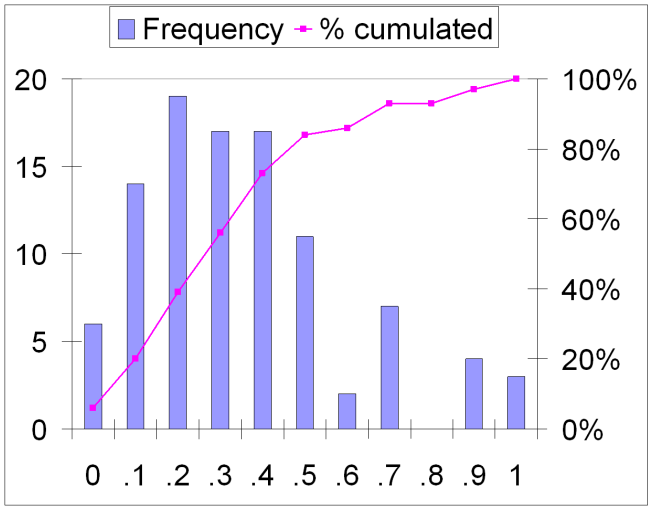
Now, we want to figure out the nature of the mistakes, if these over-detected polygons are just some tiny polygons due to noise in the raster or if they represent a major information altered during the process of conversion. To this aim, we pay attention to figure 16. Figure 16 shows that only 36% of the overall cost PMD_{all} is due to the well-detected polygons, hence, most of the information is accurately retrieved from the rasters and the retrieved polygons do fit precisely the Ground-Truth. On the opposite 64% of the mistakes are attributable to the wrongly detected polygons. Figure 16 strengthens the idea that anomalies are caused by the over-acceptation policy of the automatic application.

In another step, we aim at assessing how much manual work has to be made to correct the automatically vectorized polygons. A fact observed from figure 17 is that 54% of the MED_{all} mistakes are caused by the operations to be made when correcting the polygons MED_{tp}. A non-negligible part of the errors are caused by the corrections to be made to fit in the ground truth. An explanation could be a fragmentation phenomenon; many noisy strokes are broken into small pieces during the polygonal approximation process.

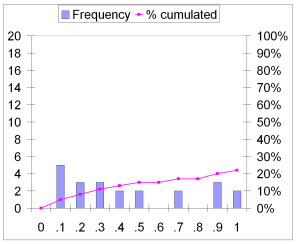
The rest of the errors, that is to say the MED_{md} values, is mainly due to an intensive use of the deletion operator in order to remove the over-detected polygons.

Finally, based on a common work with the historians Helene Noizet and Laurent Costa[4], an algorithm with a combined index (PMD_{all} + MED_{all}) of 0.70 or less may be considered good with respect to human vision

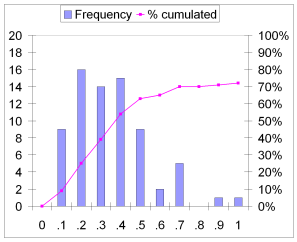
[4] Members of LAboratoire de Médiévistique Occidentale de Paris (LAMOP). UMR 8589 CNRS / UNIVERSITÉ PARIS 1 Panthéon Sorbonne



(a) η_{md}

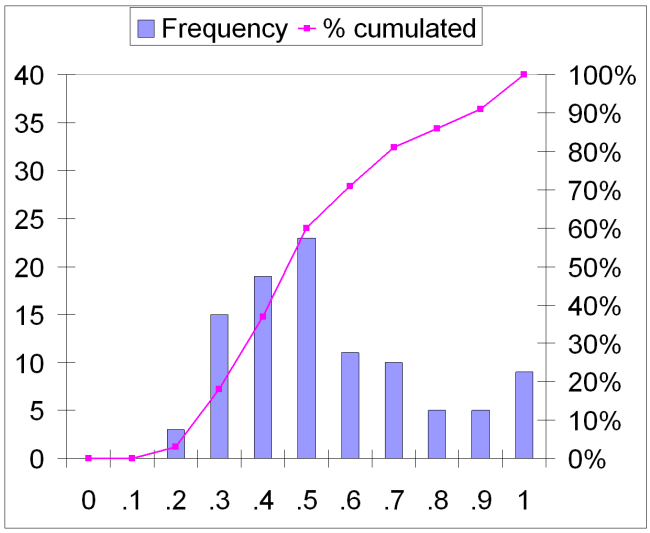


(b) η_{fn}

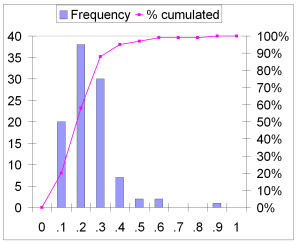


(c) η_{fp}

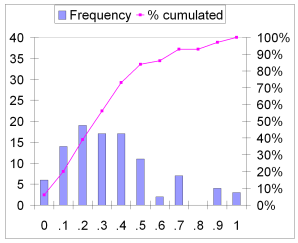
Fig. 15. Histogram of η. The mean value $\overline{\eta_{md}} = 0.31$ and it can be broken down in two parts, $\overline{\eta_{fp}} = 0.22$ and $\overline{\eta_{fn}} = 0.09$.



(a) PMD_{all}



(b) PMD_{tp}



(c) PMD_{md}

Fig. 16. Histogram of PMD. The mean value $\overline{PMD_{all}} = 0.5$ and it can be broken down in two parts, $\overline{PMD_{tp}} = 0.18$ and $\overline{PMD_{md}} = 0.32$.

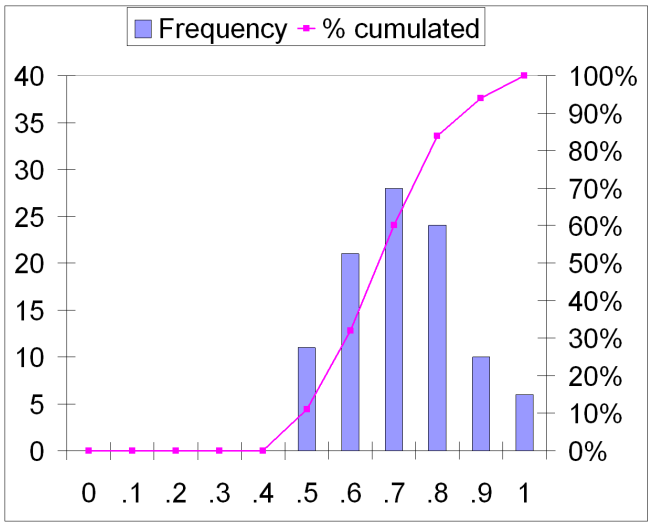
evaluation. However, more work should make use of this protocol on a series of algorithms and degraded drawings to obtain an objective assessment on commonly accepted criteria.

Inter-indices correlation: A correlation matrix is built from the data series of indices (illustrated in figure 18(c)), the Pearson correlation is 1 in the case of an increasing linear relationship, -1 in the case of a decreasing linear relationship, and some value between -1 and 1 in all other cases, indicating the degree of linear dependence between the variables. The closer the coefficient is to either -1 or 1, the stronger the correlation between the variables. This matrix aims to compare the different quality measures between them. A matrix is not expressive enough, so, a 256 shades of grey image is generated to express its substantial meaning in a 2D representation, called image of correlations (figure 18(b)). In addition, the matrix of scatterplots between the different measures of quality is given (figure 18(a)). From these data representations, a straightforward remark deals with the proportional behavior of the η_{tp} and η_{md}, which are closely coupled and share the same information. On the other hand, there is no evident relation between MED and the η measures, the Pearson correlation coefficient between these two series is not indicative enough, nevertheless, the coefficient is low enough (0.60) to indicate no significant redundancy of information. Finally, a clear tendency appears between PMD and MED, it reveals a low correlation (0.24) between PMD and MED. A situation of independence between the two series can be accepted. These variables really express two different kinds of information. They represent original viewpoints on the underlying problem.

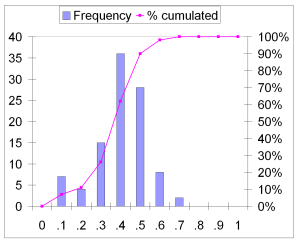
4 Conclusion and perspectives

In this paper, we defined a protocol for performance evaluation of polygon detection algorithms. A discussion between the proposed protocol and the literature is also presented. As a consequence, our protocol is positioned as an extension of prior works, an extension at polygon level. In this way, it is closer to the semantic level and closer to objects handled by humans. Former benchmarks only include synthetic images with image degradation but we completed these artificial samples by real images with manually created ground-truth. Gathering real data to test and comparing graphics recognition systems is very time consuming that is why we propose our data set to the community.

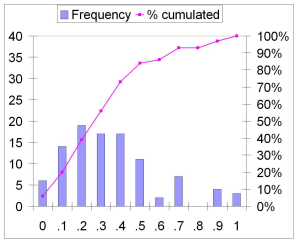
Our contribution is two-fold, an object mapping algorithm to roughly locate errors within the drawing, and then a cycle graph matching distance that depicts the accuracy of the polygonal approximation. Both were theo-



(a) MED_{all}

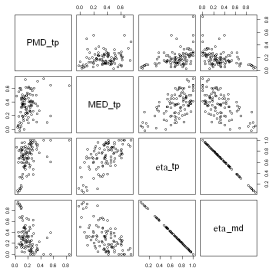


(b) MED_{tp}

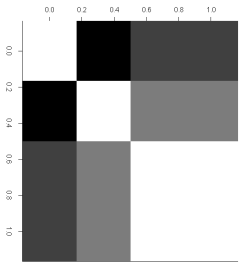


(c) MED_{md}

Fig. 17. Histogram of MED. The mean value $\overline{MED_{all}} = 0.66$ and it can be broken down in two parts, $\overline{MED_{tp}} = 0.36$ and $\overline{MED_{md}} = 0.30$.



(a) Scatterplots



(b) Image of the correlation matrix

	PMD_{tp}	MED_{tp}	η_{tp}	η_{md}
PMD_{tp}	1.0000000	0.2411222	0.4311698	-0.4311698
MED_{tp}	0.2411222	1.0000000	0.6088518	-0.6088518
η_{tp}	0.4311698	0.6088518	1.0000000	-1.0000000
η_{md}	-0.4311698	-0.6088518	-1.0000000	1.0000000

(c) Correlation matrix

Fig. 18. (a) Scatterplots of the proposed indices; (b) Image of the inter-indices correlation matrix , the lighter is the shade of grey, the higher is the correlation coefficient; (c) Correlation matrix.

retically defined and adapted to the performance evaluation of polygonized documents. Especially, cost functions were reconsidered, using a set distance for the polygon matching distance (PMD) and defining particular edit costs for the graph matching method.

The proposed protocol is objective and comprehensive, both detection and false alarm rates are considered. By stress testing a given system, we demonstrated that our protocol can reveal strengths and weaknesses of a system. The behavior of our set of indices was analyzed when increasing image degradation.

The results presented in figure 16 and 17 indicate that the proposed protocol reflects polygon detection and approximation performance accurately. In figure 18, the statistical tests demonstrated that the two proposed measures offer different kinds of information.

We have also confronted our measures of quality to a human-based evaluation. However, more work should be done in this respect to obtain an objective assessment on commonly accepted criteria.

The protocol is designed for polygons but may also be extended to other line shapes by completing the graph representation to connected vectors instead of searching for cyclic polygons. In this context, the PMD would not have to be modified at all. The MED which is representative of the manual effort to be made to correct mistakes engendered by a R2V system is envisaged through the graph matching question in terms of basic edit operations (addition, deletion, substitution). The graph formalism confers to the approach a more generic nature and opens the way to future works on more complex objects. This graph-based viewpoint could be the container of a wider range of entities. Instead of focusing on polygon items, a given element could be constituted of all connected segments to form a more complex structure than a polygon while the entire principle would remain unchanged. The graph representation is an open way to a more global paradigm, the object matching question. This could change the scope of our performance evaluation tool to the direction of object spotting.

References

1. David Byrnes. Raster-to-vector comes of age with AutoCAD Release 14. *CADALYST*, pages 48–70, 1997.
2. R Kasturi and K Tombre (eds.). Graphics Recognition: Methods and Applications. *First International Workshop, University Park, PA, USA, August 1995, Selected papers published as Lecture Notes in Computer Science*, 1072, 1996.
3. B Kong, I T Phillips, R M Haralick, A Prasad, and R Kasturi. A benchmark: performance evaluation of dashed-line detection algorithms. *Graphics recognition methods and applications (Lecture Notes in Computer Science)*, 1072:270–285, 1996.
4. D Dori, L Wenyin, and M Peleg. How to win a dashed line detection contest. *Graphics Recognition Methods and Applications, Lecture Notes in Computer Science, Springer*, 1072, 1996.
5. A Chhabra and I Phillips. The Second International Graphics Recognition Contest - Raster to Vector Conversion: A Report. *Graphics Recognition: Algorithms and*

Systems, Lecture Notes in Computer Science, Springer, 1389, 1998.
6. I Phillips, J Liang, A Chhabra, and R Haralick. A Performance Evaluation Protocol for Graphics Recognition Systems. *Graphics Recognition: Algorithms and Systems, Lecture Notes in Computer Science, Springer*, 1389, 1998.
7. I Phillips and A Chhabra. Empirical Performance Evaluation of Graphics Recognition Systems. *IEEE Transaction of Pattern Analysis and Machine Intelligence*, 21, No 9:849–870, 1999.
8. Liu Wenyin and Dov Dori. A protocol for performance evaluation of line detection algorithms. *Machine Vision and Applications*, 9(5):240–250, 1997.
9. R M Haralick. Performance characterization in image analysis thinning, a case in point. *Pattern Recogn Lett*, 13:5–12, 1992.
10. S Lee, L Lam, and C Y Suen. Performance evaluation of skeletonization algorithms for document image processing. *Proceedings of the First International Conference on Document Analysis and Recognition*, 1:260–271, 1991.
11. L Lam and C Y Suen. Evaluation of thinning algorithms from an OCR viewpoint. *Proceedings of the Second International Conference on Document Analysis and Recognition*, 1:287–290, 1993.
12. M Y Jaisimha, R M Haralick, and D Dori. A methodology for the characterization of the performance of thinning algorithms. *Proceedings of the Second International Conference on Document Analysis and Recognition*, 1:282–286, 1993.
13. L P Cordella and A Marcelli. An alternative approach to the performance evaluation of thinning algorithms for document processing applications. *Graphics recognition methods and applications (Lecture Notes in Computer Science)*, 1072:13–22, 1996.
14. R Kasturi, S T Bow, W El-Masri, J Shah, J R Gattiker, and U B Mokate. A system for interpretation of line drawings. *IEEE Trans Pattern Anal Machine Intell*, 17:978–992, 1990.
15. V Nagasamy and N Langrana. Engineering drawing processing and vectorization system. *Comput Vision Graphics Image Process*, 49:379–397, 1990.
16. A J Filipski and R Flandrena. Automated conversion of engineering drawings to CAD form. *Proc IEEE*, 80:1195–1209, 1992.
17. L Boatto. An interpretation system for land register maps. *IEEE Computer*, 25(7):25–32, 1992.
18. P Vaxiviere and K Tombre. Celestin: CAD conversion of mechanical drawings. *IEEE Comput*, 25:46–54, 1992.
19. D Dori. Vector-based arc segmentation in the machine drawing understanding system environment. *IEEE Trans Pattern Anal Machine Intell*, 17:959–971, 1995.
20. D Dori, Y Liang, J Dowell, and I Chai. Spare pixel recognition of primitives in engineering drawings. *Machine Vision Appl*, 6:79–82, 1993.
21. Rangachar Kasturi and Karl Tombre, editors. *Graphics Recognition, Methods and Applications, First International Workshop, University Park, PA, USA, August 10-11, 1995, Selected Papers*, volume 1072 of *Lecture Notes in Computer Science*. Springer, 1996.
22. Graphics Recognition, Algorithms and Systems, Second International Workshop, GREC'97, Nancy, France, August 22-23, 1997, Selected Papers. In Karl Tombre and Atul K Chhabra, editors, *GREC*, volume 1389 of *Lecture Notes in Computer Science*. Springer, 1998.
23. Atul K Chhabra and Dov Dori, editors. *Graphics Recognition, Recent Advances, Third International Workshop, GREC'99 Jaipur, India, September 26-27, 1999, Selected Papers*, volume 1941 of *Lecture Notes in Computer Science*. Springer, 2000.
24. O Hori and D S Doermann. Quantitative measurement of the performance of raster-to-vector conversion algorithms. *Graphics recognition methods and applications (Lecture Notes in Computer Science)*, 1072:57–68, 1996.
25. Robert W Floyd. Nondeterministic Algorithms. *J. ACM*, 14(4):636–644, 1967.
26. H Bunke. On a relation between graph edit distance and maximum common subgraph. *Pattern Recognition Letters*, 18(9):689–694, 1997.
27. Horst Bunke and Kim Shearer. A graph distance metric based on the maximal common subgraph. *Pattern Recognition Letters*, 19(3-4):255–259, 1998.
28. E M Arkin, L P Chew, D P Huttenlocher, K Kedem, and J S B Mitchell. An Efficiently Computable Metric for Comparing Polygonal Shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(3):209–216, 1991.
29. J Lladós, E Marti, and J J Villanueva. Symbol Recognition by Error-Tolerant Subgraph Matching between Region Adjacency Graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23:1137–1143, 2001.
30. H Bunke and G Allermann. Inexact graph matching for structural pattern recognition. *Pattern Recognition Letters*, 1:245–253, 1983.
31. A Sanfeliu and K Fu. A distance measure between attributed relational graphs for pattern recognition. *IEEE Transactions on Systems, Man, and Cybernetics*, (Part B) 1:353–363, 1983.
32. Y T Tsay and W H Tsai. Model-guided Attributed String Matching by Split-and-merge for Shape Recognition. *International Journal on Pattern Recognition and Artificial Intelligence*, 3(2):159–179, 1989.
33. Ernest Valveny and Philippe Dosch. Symbol Recognition Contest: A Synthesis. *Graphics Recognition*, pages 368–385, 2004.
34. E Valveny, P Dosch, Adam Winstanley, Yu Zhou, Su Yang, Luo Yan, Liu Wenyin, Dave Elliman, Mathieu Delalandre, Eric Trupin, Sébastien Adam, and Jean-Marc Ogier. A general framework for the evaluation of symbol recognition methods. *International Journal on Document Analysis and Recognition*, 9(1):59–74, March 2007.
35. Mathieu Delalandre, Ernest Valveny, Tony Pridmore, and Dimosthenis Karatzas. Generation of synthetic documents for performance evaluation of symbol recognition & spotting systems. *International Journal on Document Analysis and Recognition*, page Online first, 2010.
36. Philippe Dosch and Ernest Valveny. Report on the Second Symbol Recognition Contest, 2006.
37. T F Cootes, C J Taylor, D H Cooper, and J Graham. Active shape models—their training and application. *Comput. Vis. Image Underst.*, 61(1):38–59, 1995.
38. D. GHOSH and A. P. SHIVAPRASAD. An analytic approach for generation of artificial hand-printed character database from given generative models. *Pattern recognition*, 32(6):907–920, 1999.
39. E Valveny and E Marti. A model for image generation and symbol recognition through the deformation of lineal shapes. *Pattern Recognition Letters*, 24(15):2857–2867, 2003.

40. Robert M Haralickt, Henry S Baird, and David Adihan. Document Degradation Models: Parameter Estimation and Model Validation, June 2009.
41. Gabriella Sanniti di Baja and Edouard Thiel. Skeletonization algorithm running on path-based distance maps. *Image and Vision Computing*, 14(1):47–57, 1996.
42. Karin Wall and Per-Erik Danielsson. A fast sequential method for polygonal approximation of digitized curves. *Comput. Vision Graph. Image Process.*, 28(3):220–227, 1984.
43. Romain Raveaux, Jean-Christophe Burie, and Jean-Marc Ogier. A colour text/graphics separation based on a graph representation. *19th International Conference on Pattern Recognition (ICPR)*, pages 1–4, 2008.
44. Romain Raveaux, Jean-Christophe Burie, and Jean-Marc Ogier. Object Extraction from Colour Cadastral Maps. *IAPR International Workshop on Document Analysis Systems*, 0:506–514, 2008.
45. Hervé Locteau, Romain Raveaux, Sébastien Adam, Yves Lecourtier, Pierre Héroux, and Éric Trupin. Approximation of Digital Curves using a Multi-Objective Genetic Algorithm. In *18th International Conference on Pattern Recognition (ICPR)*, pages 716–719, Washington, DC, USA, 2006. IEEE Computer Society.
46. Alfredo Ferreira Jr, Jr. Manuel, J Fonseca, and Joaquim A Jorge. Polygon Detection from a Set of Lines. In *In Proceedings of 12 o Encontro Português de Computação Gráfica (12th EPCG)*, pages 159–162, 2003.
47. R Raveaux, J.-C. Burie, and J.-M. Ogier. A Colour Document Interpretation: Application to Ancient Cadastral Maps. *Document Analysis and Recognition, International Conference on*, 2:1128–1132, 2007.

Romain Raveaux received the B. Eng. and M.Sc. degrees in Computer Science and Information Engineering from the University of Rouen, France, in 2004 and 2006, respectively. He is currently a Ph.D. student in the department of Computer Science at the University of La Rochelle, France. His research interests include algorithm and data structure for image retrieval, data mining and graph-based representation.

Jean-Christophe Burie received his Ph.D. degree in Automatic Control Engineering and Industrial Data Processing from University of Lille, France, in 1995. During his thesis (1993 - 1995), he worked on stereovision algorithms for obstacle detection in the framework of the European Project EUREKA-Prometheus. He was a research fellow in the Department of Mechanical Engineering for Computer-Controlled Machinery, Osaka University, Japan from 1995 to 1997 in the framework of the Lavoisier Program of the French Foreing Office. Since 1998, Jean-Christophe Burie is assistant professor in the Computer Science Department at La Rochelle University. His current research interests include computer vision, color image processing, pattern recognition.

Jean-Marc Ogier received his Ph.D. degree in Computer Science from the University of Rouen, France, in 1994. During his thesis, he worked on cadastral maps recognition and was involved in several research projects. From 1994 to 2000, he was an Assistant Professor in Computer Engineering at the University of Rennes, and at the University of Rouen. In 2000, he obtained his advanced Ph.D. (Habilitation) in Computer Science from the University of Rouen for his work in the fields of graphic recognition. Since 2001, he has been a full Professor in the university of La Rochelle. His present research interests deal with graphic recognition, a more specifically graphic document indexing. He is the leader of the research team i-Medoc of the L3I Laboratory, and manages several national (and european research projects (MADONNE, NAVIDOMASS, RECONOMAD, ...) He is deputy director of the GDR I3 of the French CNRS and is Vice-Chair of the Technical 10 (Graphic Recognition) of the International Association for Pattern Recognition.