Morphological reconstruction of semantic layers in map images

Alexey Podlasov
Eugene Ageenko
Pasi Fränti
University of Joensuu
Department of Computer Science
Box 111
FIN-80101 Joensuu
Finland

Abstract. *Map images are composed of semantic layers depicted in arbitrary color. Color separation is often needed to divide the image into layers for storage and processing. Separation can result in severe artifacts because of the overlapping of the layers. In this work, we introduce a technique to restore the original semantic layers after the color separation. The proposed restoration technique improves compression performance of the reconstructed layers in comparison to the corrupted ones when compressed by lossless algorithms such as International Communication Unit (ITU) Group 4 (TIFF G4), Portable Network Graphics (PNG), Joint Bi-level Image experts Group (JBIG), and context tree method. The resulting technique also provides good visual quality of the reconstructed image layers, and can therefore be applied for selective layer removal/ extraction in other map processing applications, e.g., area measurement.* © 2006 SPIE and IS&T. [DOI: 10.1117/1.2178188]

1 Introduction

Currently, there exist various services delivering map imagery content to the user. For example, real-time map imaging applications provide users with a view of a geographical map for the area surrounding the user's location. The location can be obtained using a global positioning service (GPS), mobile positioning service (MPS), or other analog services. It could also be weather, traffic, pollution, or any other kind of map. The imagery data are usually obtained from a digital spatial library,[1] and transmitted via the network to the user's device such as a pocket computer (PDA), mobile phone, or desk-top terminal.

A typical map image consists of a set of semantic layers, each containing data with distinct semantic content, each depicted with its own color, e.g., black roads, brown elevation lines, blue water areas, yellow fields, etc. Regardless of the semantic nature, typical maps need only a few color tones to represent the layers, but high spatial resolution for representing details. Let us call these images multilayer map images.

We consider topographic images from the National Land Survey of Finland (NLS) topographic database, in particular the basic map series 1:20,000.[2] The images consist of a set of semantic layers, each containing data with distinct

semantic content, such as roads, elevation lines, infrastructures, state boundaries, and water areas. The layers are combined and displayed to the user as a generated color image, in which the data of each type are depicted using their own color. These images consist of the following semantic layers: basic (roads, contours, labels, and other topographic data), elevation lines (thin lines representing elevations levels), waters (solid regions and polylines representing water areas and ways), and fields (solid polygonal regions) (see Fig. 1).

The original map data are usually stored in vector format on a server-side database. Each semantic layer is stored separately. As the user's request arrives, the server prepares part of the data and transmits it to the user in raster format, since raster images are easier to handle on a client-side device. Using a vector format requires special software developed for vector map image processing, then the processing of raster images is a standard feature of almost any mobile terminal. Raster format is also often used for digital publishing on the web or CDs.

When producing a raster map image, map layers of different semantic nature are combined together overlapping each other in a predefined order. This image is well suited for user observation, but it is less appropriate for further processing, since the layer structure has been corrupted when the raster of the map image was produced. For ex-
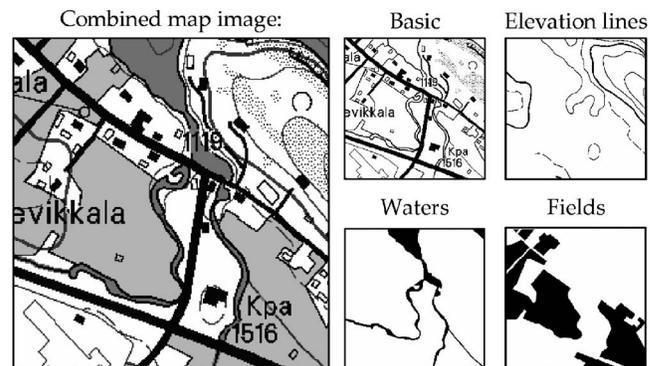

Fig. 1 Illustration of a multilayer map image from the NLS Topographic database.

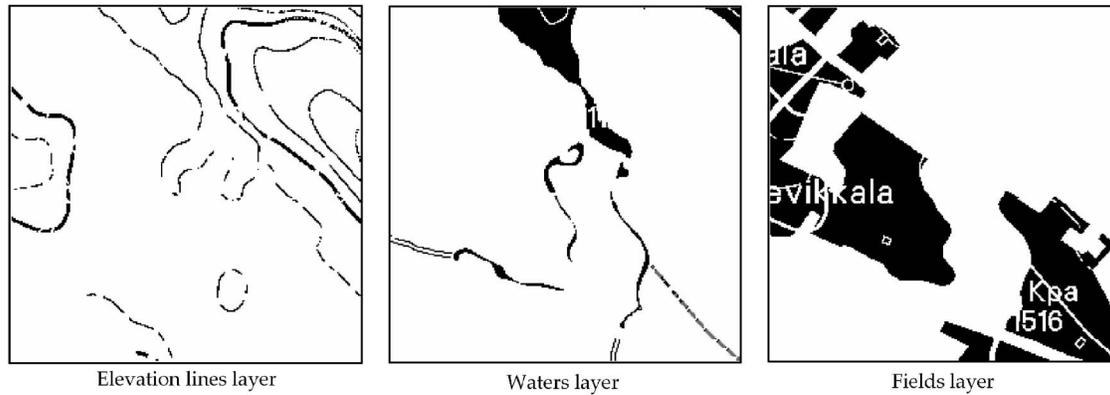| Elevation lines layer | Waters layer | Fields layer |

Fig. 2 Corrupted layers due to the color separation.

ample, when one needs to calculate the area of the fields, or the length of coastline, the layer must be extracted from the color raster image through color separation. During this process, the map image is divided into binary layers, each representing one color in the original image. The main problem of this approach is that the separation introduces severe artifacts in places where the information of different layers overlap each other (see Fig. 2). The holes on the fields caused by overlapping letters are a typical example of these artifacts. The presence of the artifacts can make the color separated layer useless for many map processing tasks.

Moreover, the problem also affects the compressibility of the images. Though the raster image could be compressed with any existing lossless compression algorithm, it has been shown that the best compression results can be achieved if the image is decomposed into binary semantic layers, which are consequently compressed by algorithms designed to handle binary data (e.g., JBIG).[3] The artifacts of the color separation, however, affect the statistical properties and consistency of the layers, and result in degraded compression performance in comparison to the original ones. This is apparent especially in applications requiring the use of mobile hardware such as mobile phones or pocket computers. For example, a single map sheet of 10 $\times$ 10 km^2 is represented by a single map image of 5000 $\times$ 5000 pixels. Larger image size also takes a longer time to transmit. For example, 10-sec transmission via a GPRS channel with bandwidth 45 kb/sec results in at most 54 kB of image data. This corresponds to only about 500 $\times$ 500 pixels for a four-layer map image.

The problems mentioned led us to develop an algorithm for the reconstruction of the corrupted layers of map images. The proposed algorithm approximates the original layer structure existing before the color combination by repairing the corrupted layers as close to the original ones as possible. A natural restriction for the reconstruction technique is that the color combination of the reconstructed layers should be equal to the originally received raster map image.

The goal of image restoration is to reconstruct the original image before degradation.[4] The reconstruction involves a criterion for measuring the quality of the desired result. For our problem, we consider two criteria: image quality and image compressibility. The first criterion measures how close the reconstructed layer is to the original semantic layer. This is important for applications where visual quality is essential, or the reconstructed layer is used for processing, e.g., measuring the area of the fields. The second criterion aims to modify the corrupted layer so that its compressibility will be improved as much as possible without causing any changes to the corresponding output color image.

The artifacts appearing on the layers could be treated as noise. If we guarantee that the color map remains untouched, noise removal could be considered as a tool for improving image quality for achieving better compressibility. There are many image enhancement methods in the literature,[4–7] and various reconstruction techniques have been considered.[8–11] Statistical modeling,[12] and specific data modeling and representation techniques[13–16] have also been considered. However, noise filtering and typical image enhancement algorithms are not suitable for solving our
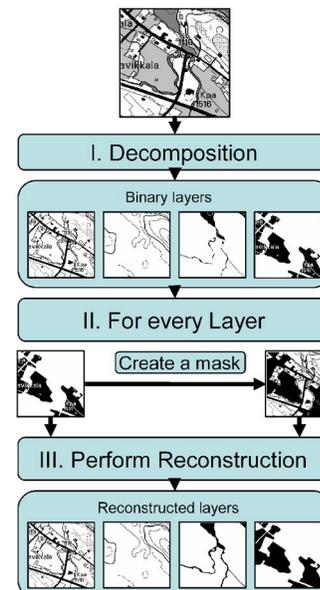

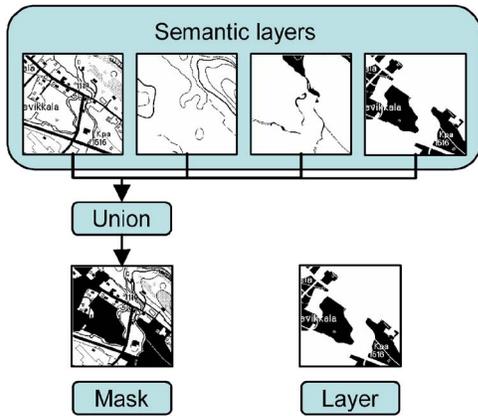**Fig. 3** Outline of the reconstruction algorithm.
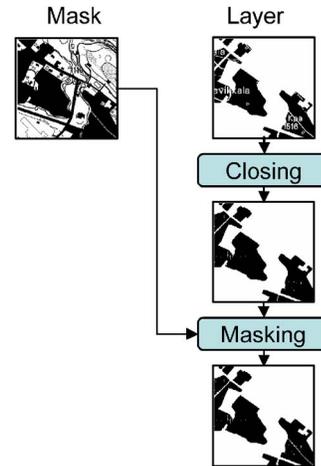
Fig. 4 Scheme for the mask creation.

Fig. 6 The block diagram of CC algorithm.

problem, because, due to their local nature, they are not able to recognize larger structures and dependencies between layers.

Therefore, we introduce a new morphological filter for layer reconstruction. We chose mathematical morphology to be the tool, mostly due to the simplicity of implementation. Morphological operators do not require sufficient computational and memory resources to be applied, which is apparent for use on mobile terminals. The benefits of the proposed filter are its capability to reconstruct semantic information in a multilayer map image, and that the original color image can always be reconstructed exactly without any loss in the quality. The effect of the filter is therefore limited only to the binary layers. The method is applicable for extraction or removal of individual layers, and for lossless compression of the map images. The method is fast and simple to implement.

The rest of the work is organized as follows. Mathematical morphology is briefly introduced in Sec. 2. In Sec. 3, we introduce two variants of the new filtering method for layer extraction, and then apply it for layer removal in Sec. 4. Empirical results are reported in Sec. 5, and conclusions drawn in Sec. 6.

2 Mathematical Morphology

Mathematical morphology refers to a branch of nonlinear image processing and analysis originally introduced by Matheron[17] and Serra,[18] and currently continuing its development.[19] This chapter gives the basic morphological definitions. In discrete binary morphology, an image space
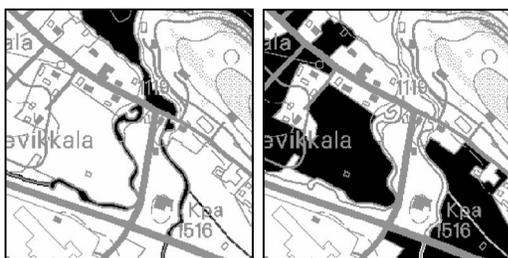

Fig. 5 Water and field layers with their masks. Object pixels from the layer are plotted by black, and the mask pixels by gray to differentiate which pixel belongs to the layer, and which pixel to the mask.

E is usually defined as $E=\mathbb{Z}^2$ (the space of all possible image pixel locations), and a binary image X as a set $X \subseteq E$. For a given set A, the reflection (or the symmetric) of A with respect to the origin, denoted as $\tilde{A}$ or $-A$, is defined by $\tilde{A} = \{-a \,|\, a \in A\}$. The power set of E, or in other words, the set of all subsets of E, is denoted as $\mathcal{P}(E)$. One of the main fundamentals of mathematical morphology is to analyze the geometrical and topological structure of an image X by "probing" the image with another small set $A \subseteq E$ called a structuring element. The choice of the appropriate structuring element depends on the particular application.

2.1 Fundamental Morphological Operators

The dilation of X by A, denoted as $\delta_A(X)$, is defined as the operator on $\mathcal{P}(E)$ given by:

$$\delta_A(x) = \bigcup_{a \in A} X_a = \{h \in E | \widetilde{A_h} \cap X \neq \varnothing \}.$$

The erosion of X by A, denoted by $\varepsilon_A(X)$, is

$$\varepsilon_A(X) = \bigcap_{a \in A} X_{-a} = \{h \in E | A_h \subseteq X\}.$$

The cardinality of set A, or the number of elements in A, is denoted by $\mathrm{card}(A)$. Let us also define the translation invariant operator $\rho_{A,n}$, called a rank operator, as follows:

$$\rho_{A,n}(X) = \{h \in E | \mathrm{card}(X \cap A_h) \geq n\}.$$

The operator $\rho_{A,n}(X)$ sets current pixels to be the foreground if the amount of foreground pixels in a neighborhood defined by the structuring element is greater than n. Otherwise, the pixel is defined as a background pixel. Since the rank operator performs similar to erosion or dilation depending on the value of the rank parameter, it is possible to treat the rank as a soft counterpart of classical erosion and dilation operators. In particular:
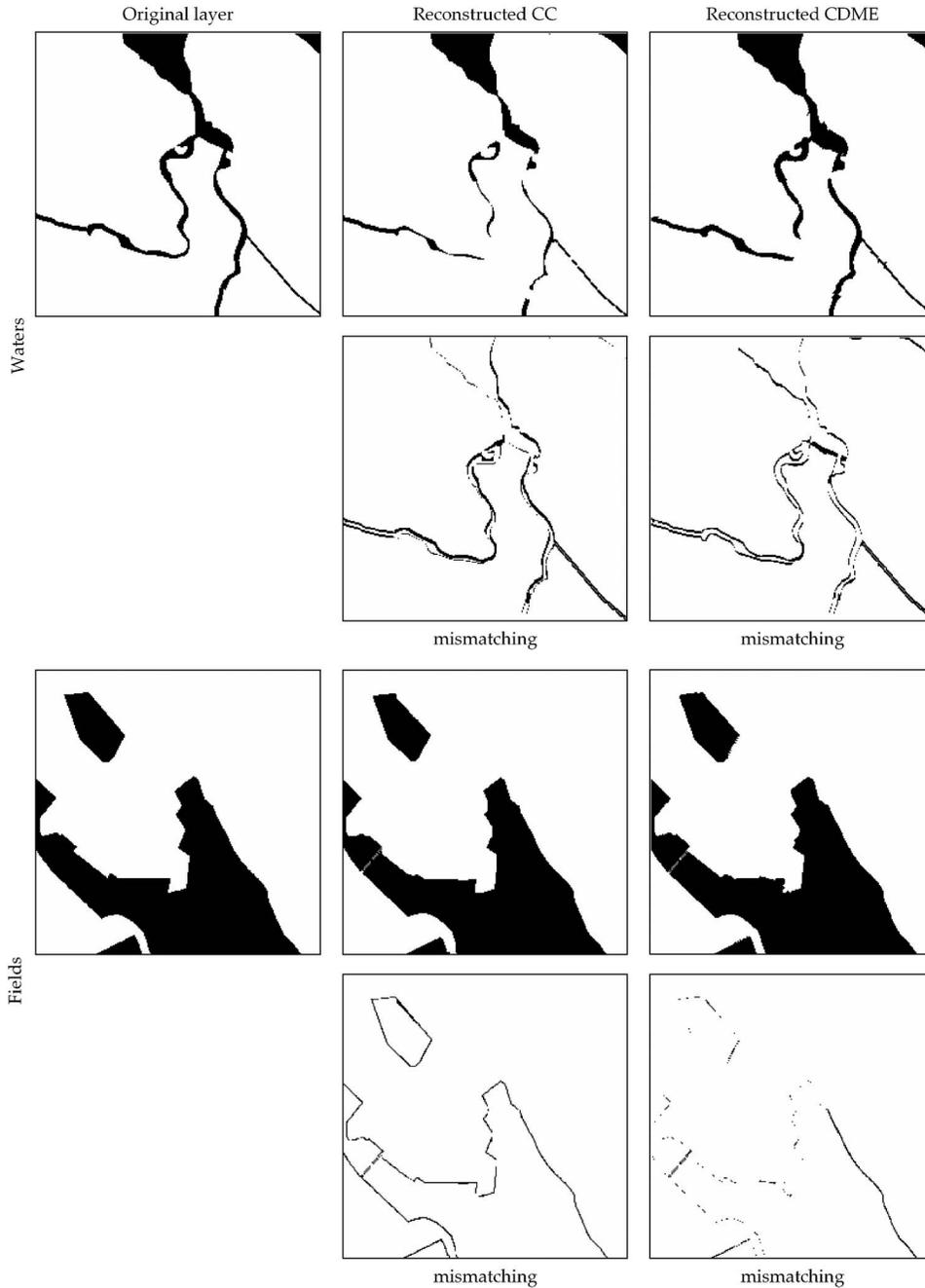
Fig. 7 Sample images for the water and field layers: original and reconstructed with CC and CDME algorithms as well as mismatching of reconstructed layer and the original.

$$\delta_A^-(X) = \rho_{A,1}(X) \text{ and } \varepsilon_A(X) = \rho_{A\,\mathrm{card}(A)}(X).$$

The operator $\alpha_A(X) = \delta_A[\varepsilon_A(X)]$ is called the (structural) opening by A. Dually, the operator $\beta_A(X) = \varepsilon_A[\delta_A(X)]$, is called the (structural) closing by A.

2.2 Conditional Operators

If an image is, say, dilated by a structuring element containing the origin, it is expanded, and the manner of the expansion depends only on the shape of the structuring element. If the dilation is successively repeated, the original image grows without bounds. Sometimes it is important to restrict the growth. This can be accomplished by using conditional operators. A common form of conditioning restricts the translations to a superset of the input image: if image A is a subset of image T, then for any operator $\psi(A)$, the operator $\psi(A|T)$ is called $\psi(A)$ conditioned relative to T and is defined as:

$$\psi(A|T) = \psi(A) \cap T.$$

The image T is usually referred to as a mask image.

$L \leftarrow$ Layer
$M \leftarrow$ Mask
$A \leftarrow$ Structuring element of dilation
$B \leftarrow$ Structuring element of erosion
  REPEAT

    $L := \delta_A(L \mid M);$

    $M := \varepsilon_B(M);$

    $M := L \cup M;$

  UNTIL Iteration criterion met

Fig. 8 Outline of the CDME algorithm.

3 Layer Reconstruction Technique

We consider two approaches. The first approach aims at maximal compression improvement for the reconstructed layers, and the second at more accurate restoration of the original semantic layers.[20] In the first approach, we simply try to minimize the storage size of the layers, which is essential for map storage systems. In the second approach, we try to produce layers that are as close to the original semantic layers as possible. The resulting layers can then be used for additional map processing and analysis. Following the underlying principles behind the previous two approaches, we have designed two reconstruction algorithms referred to further as conditional closing (CC) and conditional dilation with mask erosion (CDME).

Both algorithms have the same structure, consisting of three main steps as outlined in Fig. 3. At the first step, the color map (scanned or obtained from the third party source) is decomposed into a set of binary layers by a color separation process. This is done so that each layer represents one color in the original image.[3] Then, according to the predefined layer order, a conditioning mask is created for every layer for restricting the reconstruction of the layers to be equal to the original color image. Finally, the actual reconstruction is performed for every layer with respect to its conditioning mask.

3.1 Conditioning Mask

Further, we denote a layer image as L; when we talk about some particular layer, we denote it as L_k, $1 \le k \le N$, where N is the total number of layers in a map image. The requirement that the composition of reconstructed layers should be
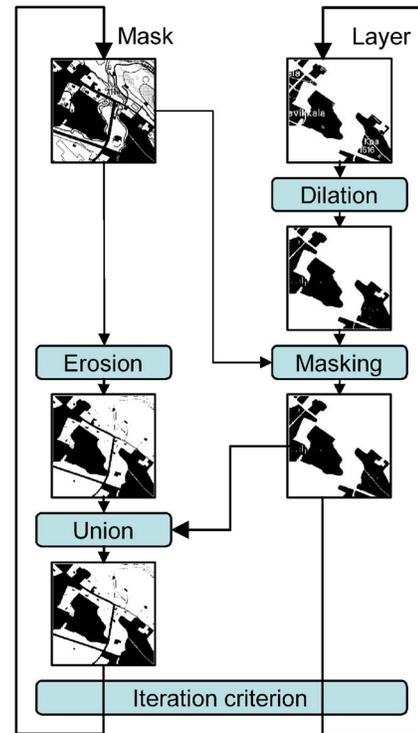

Fig. 9 Block diagram of the CDME algorithm.

identical to the initial color map can be met by conditioning the operator $\psi(L_k)$ on the mask M_k, which defines the region where changes of the layer content are allowed. The requirement of keeping the reconstructed color map identical to the original one leads to the fact that the restoration operator must not remove pixels that are already present in the corrupted layer. It can only add pixels to a layer, so that the condition

$$L_k \subseteq \psi(L_k \mid M_k),$$

is met. The conditioning mask defines the set of pixels that are allowed to change value in the restoration, so that the combination of the restored layers would be kept untouched. Since we assume that the order of layer overlapping is predefined, the mask for every layer is computed as the union of all upper-laying layers,

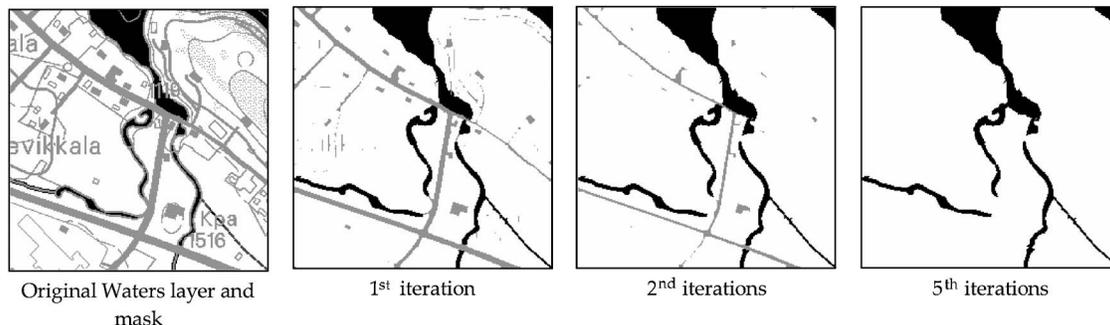| Original Waters layer and mask | 1st iteration | 2nd iterations | 5th iterations |

Fig. 10 Step-by-step illustration of the dilation with mask erosion. Object pixels from the layer are plotted by black, and the mask pixels by gray to differentiate which pixel belongs to the layer, and which pixel to the mask.
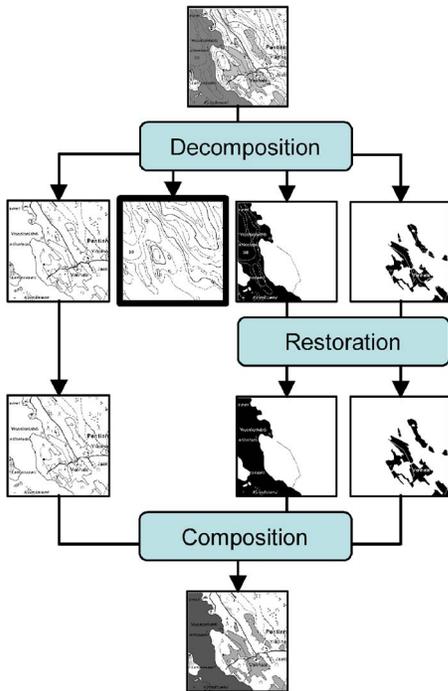
Fig. 11 Block diagram of the layer removal algorithm. Elevation lines layer to be removed is outlined with a black frame.

$$M_k := \bigcup_{j=1}^{k} L_k,$$

(see Figs. 4 and 5).

3.2 Layer Reconstruction

3.2.1 Conditional closing

Having the compression objective in mind, let us consider using a simple and effective conditional closing (CC) operator $L := \beta_A(L|M)$ to perform reconstruction. The algorithm is outlined in Fig. 6. The quality of the reconstruction in terms of compressibility strongly depends on the applied structuring element. In our experiments, we have tried out several alternatives and found that square block provides the best compression improvement. The size of the block depends on the size of the artifacts, and, for our test set 7 $\times$ 7 has been selected. Once applied successfully, the closing fills artifacts inside the objects, leaving the borders almost untouched (see Fig. 7). The main characteristics of this approach are its simplicity of implementation, and its positive effect on compression.

3.2.2 Conditional dilation with mask erosion

Although efficient in terms of compression, the CC algorithm is not as effective in approximating (restoring) the original layers. The method expands the lower layers too conservatively, whereas the color layer is typically a reduced version of the original semantic layer, due to overlapping. Therefore, we propose another algorithm, which we call conditional dilation with mask erosion (CDME), using more aggressive expansion based on dilation, and thus, aiming at a more accurate approximation of the original semantic layers.

The idea in general is to spread objects step by step and shrink the mask, too. The process is iterative: first, the spreading is performed by the dilation operator $\delta_A(X)$, and then the mask shrinking is performed by the erosion operator $\varepsilon_A(X)$. The pseudocode of the algorithm is shown in Fig. 8, and outlined in Fig. 9. The stepwise process of the iterations is illustrated in Fig. 10.

The iterative process is controlled by a stopping criterion. We have investigated two approaches: iterate until stability and iterate fixed amount of times. The first approach assumes that the iterative process will continue until the layer (and mask) converges. The convergence is guaranteed, because the erosion sequentially decreases the mask (see Fig. 10). We can therefore perform the iterations until the mask equals the layer itself.

Examination if the mask and layer are equal could be a time-consuming operation, especially if the image size is big. To avoid this, we consider the second approach by assuming that most of the artifacts are of limited size, which can be determined within the first few iterations. We therefore restrict the amount of iterations by a fixed number. For example, if we suppose that the size of an artifact is four pixels, on average, three dilations with a 3×3 block are enough for the restoration.

As with the conditional closing, an important question is the choice of an appropriate structuring element. There are
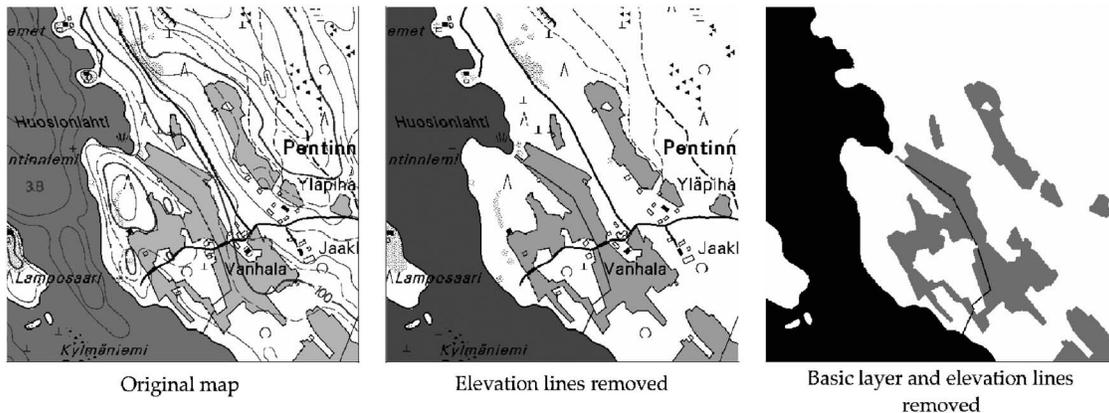

| Original map | Elevation lines removed | Basic layer and elevation lines removed |

Fig. 12 Example of the layer removal.

Table 1 Restoration of elevation layer.

Compression algorithm	Semantic layers	Corrupted layers	CDME		CC	
			Size	Imp.	Size	Imp.
PNG	825 510	808 126	811 958	−0.47%	805 774	4.30%
TIFF	460 811	481 338	464 934	3.41%	461 482	0.29%
JBIG	236 210	269 423	259 139	3.82%	253 606	6.24%
AKF2	223 555	261 386	250 820	4.04%	243 870	6.70%

two structuring elements used in the algorithm: in the object dilation and in the mask erosion. By varying the first element, we can control how fast the object expands over the mask, while varying the second element controls how fast the mask shrinks. An essential matter is the relation between the speeds of the dilation and erosion. Let A be the structuring element of dilation and B be the structuring element of erosion. We use two structuring elements: square is the 3×3 block $\{[-1,0,1] \times [-1,0,1]\}$ and cross $\{(0, -1), (1,0), (0,0), (-1,0), (0,1)\}$. We have tested three cases: objects dilating faster than mask eroding (A =square, B=cross), objects dilating slower than mask eroding: (A=cross, B=square), and the case of equal speed (A =square, B=square or A=cross, B=cross).

The speed of dilation and shrinking could also be controlled if dilation and erosion operators used in a restoration technique are replaced with a rank operator as their "soft" counterpart.[17] The rank operator is equal to the dilation operator when the rank parameter n is set to 1, and to the erosion operator when the rank parameter is equal to the cardinal number of the structuring element $[n = \text{card}(A)]$. Rank operators with rank parameters lying between these two values behave approximately like dilation or erosion operators. In other words, a rank parameter could be used to regulate the "strength" of erosion or dilation, or how fast objects shrink or expand. The case when a rank operator equals $\text{card}(A)/2$ is called a median operator.

The performance of the restoration strongly depends on the morphological structure of the layer under reconstruction. To choose the variant of the algorithm for evaluation, we examined different structuring elements and parameter values. The modification gaining the best performance is described and evaluated in Sec. 5.

4 Layer Extraction/Removal Technique

The task of layer restoration arises if there is a need for layer extraction or removal. Layer extraction is needed when one wishes to perform some specific processing over the layer, e.g., to calculate the area of the fields. Naturally, a corrupted layer could not be accepted as accurate input. A similar task is layer removal when less important layers are not needed by the map user, e.g., user driving a car does not need elevation lines, as such layers can limit map readability. To remove a layer, the restoration technique of Sec. 3 is first applied to all layers, and the color image is composed of the restored layers except for the one to be removed (see Fig. 11).

The most important feature here is the quality of the restoration, i.e., how closely the corrupted layer approximates the original layer. Moreover, in user interactive applications, the visual appearance of the reconstructed layer becomes essential. Figure 12 illustrates the effect of the removal of the basic and elevation layers.

5 Evaluation

The restoration techniques have been evaluated on a set of topographic color-palette map images. These images were decomposed into binary layers with distinctive semantic meaning identified by the pixel color on the map. The restoration algorithms have been applied for reconstruction of these semantic layers after the map decomposition process. Both the combined color map images and the binary semantic layers composing these color map images were originally available for testing. This allowed us to compare the restored images with their original undistorted counterparts.

Table 2 Restoration of water layer.

Compression algorithm	Semantic layers	Corrupted layers	CDME		CC	
			Size	Imp.	Size	Imp.
PNG	381 608	425 862	384 766	9.65%	378 484	11.13%
TIFF	167 361	357 164	168 630	52.79%	171 673	51.93%
JBIG	81 334	137 258	93 230	32.08%	95 520	30.41%
AKF2	49 230	73 107	57 370	21.53%	54 695	25.18%

Table 3 Restoration of field layer.

Compression algorithm	Semantic layers	Corrupted layers	CDME		CC	
			Size	Imp.	Size	Imp.
PNG	309 712	456 710	320 821	29.75%	313 486	31.36%
TIFF	99 622	196 456	105 388	46.36%	119 306	39.27%
JBIG	49 409	113 977	50 936	55.31%	56 950	50.03%
AKF2	5917.5	16110.5	7056.25	56.20%	6 212	61.44%

The performance of the proposed restoration techniques was evaluated according to two measures: the improvement of compression performance and the quality of the reconstruction. The first measure is relevant when dealing with map image storage, and concerns only the improvement in compressibility, regardless of how exact the reconstruction is. The second measure is relevant to applications where the reconstruction is expected to approximate the original as close as possible.

The test set consists of five randomly chosen images from the NLS Basic Map Series 1:2000, corresponding to the map sheets 431306, 201401, 263112, and 431204. Each image is of 5000×5000 pixels and consists of four binary layers. The layer names are the following:

- basic—topographic image, supplemented with communications networks, buildings, protected sites, benchmarks, and administrative boundaries
- elevation—elevation lines
- water—lakes, rivers, swamps, and water streams
- fields—agricultural areas.

5.1 Compression Performance

The evaluation examines the compression performance of the map images constructed on reconstructed layers in comparison to semantic layers (not affected by the layer separation process) and corrupted layers (result of the layer separation). The proposed algorithms were evaluated using four compression techniques: LZ (PNG), ITU Group 4 (TIFF), JBIG, and AKF2[21] (context-based compression with optimized context size and shape). For each of these compression methods, we have measured the compressed data size for the original semantic layers, for the corrupted binary layers after decomposition, and for the reconstructed layers with the two reconstruction algorithms (CC and CDME). The structuring elements in CC are 7×7 blocks; the CDME uses soft erosion and dilation with rank parameters 2 and 8, respectively.

5.1.1 Results for stand-alone layers

Tables 1–3 give the average compressed sizes of the restored elevation lines, water, and field layers, respectively. The results are the average compressed file size (size) improvement in compression ratio (imp) for semantic, corrupted, and reconstructed layers.

Evaluating the performance for the elevation-line layer we conclude that neither reconstruction technique is effective in improving the compression performance. The structure of the layer does not allow for remarkable increase, and only about 5% improvement was achieved. On the other hand, significant compression improvement is gained for water and field layers (20 to 50%, depending on the compression technique), since these layers contain a lot of closed solid regions. The holes left by letters and other artifacts were successfully filled by both algorithms. The tradeoff between the algorithms is in the computational complexity (the CC is simpler) and the quality of the restoration (the CDME has significantly better visual appeal, as shown further).

5.1.2 Overall results

In a real application, however, one cannot consider compression improvement for independent layers, but we must evaluate the compression performance altogether for all

Table 4 The average compression performance of the topographic map images based on semantic layers, color layer separation, and reconstructed color layer separation with CDME and CC restoration algorithms.

Compresson algorithm	Semantic layers		Corrupted layers		Reconstructed with CDME			Reconstructed with CC		
	Size	bpp	Size	bpp	Size	bpp	imp.	Size	bpp	imp.
PNG	2 085 871	0.66	2 149 490	0.68	2 078 254	0.66	3.31%	2 063 955	0.66	3.98%
TIFF	1 473 824	0.47	1 708 362	0.54	1 480 657	0.47	13.33%	1 483 727	0.47	13.15%
JBIG	684 978	0.21	790 257	0.25	720 185	0.23	8.87%	718 446	0.22	9.09%
AKF2	624 117	0.19	696 017	0.22	660 661	0.21	5.08%	650 191	0.20	6.58%

layers forming a map image. Table 4 illustrates the average compression performance of the test images based on semantic layers, color layer separation, and reconstructed color layer separation with CDME and CC restoration algorithms. The results are average compressed file sizes (size) computed as the sum of all compressed layers, bit rate (bpp), and improvement ratio (imp). We conclude that the proposed restoration technique achieves almost the same degree of compression of the map images as if the original semantic layer decomposition was available. Relatively low compression improvement is caused by the dominant size of the nonrestorable top-level layer basic, or the hardly restorable elevation-line layers.

5.2 Restoration Quality

This section evaluates the restoration performance of the proposed technique. By restoration quality, we mean how close the original and reconstructed layers are, with respect to some distance measure. In this work, we use normalized mean absolute error (NMAE), i.e., Hamming distance, which measures the average number of different pixel values in the original semantic layers, and in the reconstructed layers.

$$\mathrm{NMAE}(X, Y) = \frac{\sum_{j=1}^{\mathbb{H}} \sum_{i=1}^{\mathbb{W}} |x_{i,j} - y_{i,j}|}{\mathbb{H} \cdot \mathbb{W}},$$

where H and W are image dimensions.

The compression evaluation showed that the elevation layer is hardly restorable. Therefore, we do not consider it in the quality evaluation. We measured the NMAE difference between the original layers of water and field, and their reconstructed counterpart, with both CC and CDME. The same was done for the corrupted layers with respect to the original ones. These results show that the reconstructed layers are closer to the original layers than the corrupted ones. In Fig. 13, we present the total NMAE differences within the test set for each layer separately. The CDME algorithm showed better reconstruction comparing CC both for water and fields.

We evaluated the performance of the restoration by applying it to the task of area measurement. We compared the area measured over the original layer with one measured over the reconstructed and corrupted layer. The results are presented for water and field layers separately on average within the whole test set (see Table 5). Since CDME approximates layers better, its area measurements are also much closer to the original than the CC results. CDME reconstruction reduces the error of the area measurement from 15 to 20% to just about 1%.
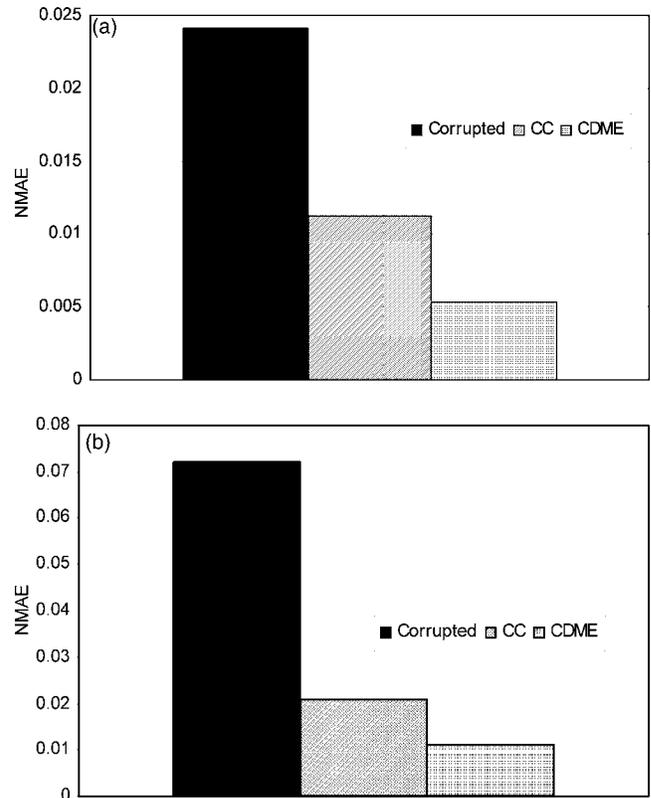

Fig. 13 The average NMAE difference with the original measured for reconstruction with CC and CDME field (a) and water (b) layers compared to corrupted ones.

6 Conclusion

We propose a technique for the restoration of binary semantic layers of map images from the corruption caused by the decomposition of the image using a color separation process. The performance of the proposed method is evaluated by improvement in compression performance and in quality of the restoration. It allows us to obtain up to 30 to 50% compression improvement for stand-alone layers and improves the total compression rate (calculated for the sum of the layers) up to 5 to 10%, depending on the compression method. Low total improvement rates are caused by the presence of non- or hardly restorable layers, such as basics and elevation.

Quality evaluation shows that restoration efficiently approximates corrupted layers to the original. The properly tuned algorithm reduces error in such applications as area measurement from 15 to 20% to about 1%. The color map

Table 5 The area (in pixels) measured over the original, corrupted, and reconstructed with CC and CDME elevation, water, and field layers.

Compression algorithm	Semantic layers	Corrupted layers		CDME		CC	
	Area	Area	Error, %	Area	Error, %	Area	Error, %
Water	10 480 893	8 678 605	17.20%	10 389 501	0.87%	9 996 454	4.62%
Field	4 267 983	3 663 960	14.15%	4 262 378	0.13%	4 057 253	4.94%

image resulting from the combination of the reconstructed layers remains identical to the original image, because all changes to the layer content are performed only within those areas that will be overlapped during composition. The method therefore affects only the separated layers, not the original color image.

References

1. "Introduction to digital libraries," in *Commun. ACM* **38**(4), 22–28, E. A. Fox, R. Akscyn, R. Furuta, and J. Leggett, Guest Eds. (1995).
2. National Land Survey of Finland, NLS, Opastinsilta 12 C, P.O. Box 84, 00521 Helsinki, Finland, see http://www.nls.fi/index_e.html.
3. P. Fränti, E. Ageenko, P. Kopyloy, S. Gröhn, and F. Berger, "Compression of map images for real-time applications," *Image Vis. Comput.* **22**(13), 1105–1115 (Nov. 2004).
4. R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd ed., Addison-Wesley, New York (2002).
5. I. Pitas and A. N. Venetsanopoulos, *Nonlinear Digital Filters: Principles and Applications*, Kluwer, Boston, MA (1990).
6. E. R. Dougherty, "Optimal mean-square n-observation digital morphological filters. Part 1: Optimal binary filters," *Comput. Vis. Graph. Image Process.* **55**, 36–54 (1992).
7. *Nonlinear Filters for Image Processing*, E. R. Dougherty and J. Astola, Eds., SPIE Optical Engineering Press, Bellingham, WA (1997).
8. D. Ting and B. Prasada, "Digital processing techniques for encoding of graphics," *Proc. IEEE* **68**(7), 757–769 (1980).
9. F. M. Wah, "A binary image preprocessor for document quality improvement and data reduction," *Proc. Intl. Conf. Acoustic, Speech, Signal Process. ICASSP'86*, pp. 2459–2462 (1986).
10. Q. Zhang and J. M. Danskin, "Bitmap reconstruction for document image compression," *Proc. SPIE* **2916**, 188–199 (1996).
11. L. Koskinen and J. Astola, "Soft morphological filters: A robust morphological filtering method," *J. Electron. Imaging* **3**, 60–70 (1994).
12. E. Ageenko and P. Fränti, "Context-based filtering of document images," *Pattern Recogn. Lett.* **21**(6,7), 483–491 (2000).
13. Z. Ping, C. Lihui, and K. C. Alex, "Text document filters using morphological and geometrical features of characters," *Proc. Intl. Conf. Signal Process. ICSP'00*, pp. 472–475 (2000).
14. T. R. Randolph and M. J. T. Smith, "Enhancement of fax documents using a binary angular representation," *Proc. Intl. Symp. Intell. Multimedia, Video Signal Process.*, pp. 125–128 (2001).
15. Q. Zheng and T. Kanungo, "Morphological degradation models and their use in document image restoration," Univ. Maryland Technical Report, LAMP-TR-065 CAR-TR-962 N660010028910/IIS9987944 (2001).
16. P. Fränti, E. Ageenko, S. Kukkonen, and H. Kälviäinen, "Using Hough transform for context-based image compression in hybrid raster/vector applications," *J. Electron. Imaging* **11**(2), 236–245 (2002).
17. G. Matheron, *Random Sets and Integral Geometry*, Wiley and Sons, New York (1975).
18. J. Serra, *Image Analysis and Mathematical Morphology*, Academic Press, London (1982).
19. H. J. A. M. Heijmans, *Morphological Image Operators*, Academic Press, Boston (1994).
20. E. Ageenko and A. Podlasov, "On the restoration of semantic data in raster map images," *IADIS Informatics Int. Conf. (I2005), IADIS Virtual Multi Conf. Computer Sci. Info. Sys. (MCCSIS 2005)*, pp. 334–342.
21. E. Ageenko, P. Kopylov, and P. Fränti, "On the size and shape of multi-level context templates for compression of map images," *Proc. Intl. Conf. Image Process. ICIP'01* **3**, 458–461 (2001).

Alexey Podlasov received his MSc degree in applied mathematics from Saint-Petersburg State University, Russia, in 2002, and his MSc degree in computer science from the University of Joensuu, Finland, in 2004. Currently, he is a doctoral student in computer science at the University of Joensuu. His research topics include processing and compression of map images.

Eugene Ageenko received his MSc degree in applied mathematics and software engineering in 1995 from the Moscow State University, Russia, and his PhD degree in computer science in 2000 from the University of Joensuu, Finland. Currently, he is a senior assistant for the Computer Science Department of the University of Joensuu, Finland. His research interests include line-art image processing and compression, mobile imaging, and location-based systems. He is a member of IEEE and ISO/IEC JTC 1/SC 29/WG1 (JBIG).

Pasi Fränti received his MSc and PhD degrees in computer science in 1991 and 1994, respectively, from the University of Turku, Finland. From 1996 to 1999 he was a postdoctoral researcher of the Academy of Finland. He has been a professor at the University of Joensuu, Finland since 2000. His primary research interests are in image compression, clustering, and speech technology.