1 Introduction

1.1 Raster-to-vector conversion

Vectorization (raster-to-vector conversion) consists of analyzing a raster image to convert its pixel representation to a vector representation The basic assumption is that a vector representation is more suitable for the interpretation of the image, which typically is a scanned graphical document (map, scheme, technical or construction drawing).

Traditionally the vectorization of maps and other documents has been performed manually on a digitizing table using a mouse. With the appearance of document scanners the manual procedure was modified: the scanned map and the result are displayed on a high resolution screen in an overlay, but the vectorization is done manually using mouse. Recently, with the development of appropriate software, the procedure of vectorization has become semi- or fully automatic.

To support the transition from manual processing to fully automatic vectorization, a lot of problems in image processing, image analysis and pattern recognition have had to be solved during the last 30 years. Recently commercial software has became available on market. But at the same time, we cannot say that all the problems have been completely solved. Tombre *et al*. wrote [264]: ”*Actually, the methods do work, but none of them is perfect. ... Although these methods yield good results, they all have their specific weakness, so that we cannot say that perfect raster-to-vector conversion is available*”.

The future progress in this area is mostly connected with further development of the methods and techniques of vector image analysis and interpretation. Along with this, future progress is impossible without improvement of the basic (so called low-level) algorithms in image processing, including algorithms for image preprocessing and vectorization. This is caused by the following reasons. The first reason is that, with continuous increasing computer resources (speed, memory), the practical demands are also growing. For example, scanned document of E-size (34”×44”) at 200 dpi resolution generates 80 Mb of raw data; at 2000 dpi resolution 800 Mb are needed [3]. Using powerful workstations or special purpose hardware for solving practical tasks is expensive for most customers. Development of more efficient

algorithms and methods for processing of large data with an ordinary desktop is still an important problem.

The second reason is introducing of new types of computers and devices with more constrained power and memory resources (laptops, palmtops, pocket PCs, wearable computers, communicators, mobile phones). Now image for processing can be acquired with handy scan, digital camera or received via Internet and using wireless connection. For example, user can scan Chinese or Japanese hieroglyphs on the street with pocket device with a digital camera. Then the text will be recognized by the device and translated to the user. Another example of a new area of vectorization module is sketchpad interface for palmtops and wearable computers.

Development of new algorithms and more efficient implementations for existing algorithms for the problem in question is an important task for researchers. With the term "efficient algorithms" we mean algorithms that have good a balance between quality and time performance.

1.2 Polygonal approximation

The problem of approximating a given two-dimensional piecewise linear curve by another coarser one is of fundamental importance [53]. In our case, the problem is important because of two reasons. At first, approximation of curves is an essential part of the vectorization procedure. At second, the approximation of digital curves is widely used for vector data processing (data reduction, compression, map simplification) in digital cartography, GIS applications, and CAD systems.

For the last 30 years, the problem of polygonal approximation has been studied by many researchers. The approximation problem can be solved with optimal methods, but as Heckbert and Garland wrote [101]: "*Optimal simplification typically has quadratic or cubic cost, making it impractical for large inputs*". Heuristic algorithms of lower complexity have been developed for vectorization and vector data processing in practical applications. One can count up to a dozen different heuristic approaches to polygonal approximation whereas the number of proposed algorithms exceeds one hundred.

On the one hand, existing optimal algorithms are too slow to be practical. On the other hand, the fast heuristic algorithms lack optimality. The main goal of our research has been to develop efficient algorithms for polygonal approximations, which are very close to optimal (or even optimal), and have linear or near-linear time complexity. In other words, the developed algorithms have to be as fast as

heuristic algorithms and yet provide results that are very close to those of optimal algorithms.

1.3 Structure of the Thesis

In Chapter 2 we study the problem of raster-to-vector conversion in general, and then we consider the problem of processing large size input data, paying attention to efficient implementation of some low-level processing algorithms including binarization and skeletonization. We also present binary noise filtering methods, paying special attention to methods that use vectorization.

In Chapter 3, we explore more deeply the polygonal approximation of digitized curves and vector data. Our goal is to develop efficient algorithms in this area including solutions for *min*-ε and *min*-# *problems*, approximation of closed contours, approximation of planar curves and 3-D paths, approximation of single and multiple curves, approximation of digitized curves for vectorization, vector data reduction and map simplification.

1.4 Summary of the publications

In the first paper (P1), we study the problem of vectorization of grayscale images of large size. The main purpose of the algorithms developed is processing of image of very large size. First, we offer algorithm for locally adaptive binarization of large grayscale images. Second, we present a fast implementation of a skeletonization algorithm for images of very large size. The algorithm was used in the development of a parallel implementation in the paper **P3**. With this algorithm very large images can be processed by a single run of reading of the image file, instead of several time-consuming reading of the file in forward and backward directions. Third, we present an algorithm for the analysis of vectors to define the most informative part of the obtained vector data. The developed algorithms and software were used in practice for vectorization as well for solving problem of noise filtering presented in publication **P2**.

In paper **P1**, the author developed the main principles and algorithms for the proposed raster-to-vector conversion system and implemented the pre-processing and vectorization parts. He is the principal author of the paper. The other two authors took part in the implementation of algorithms.

In the second paper (P2), we explore the problem of noise filtration in binary images of certain type. Normally, noise filtration is a part of the raster-to-vector conversion procedure. In the publication we use vectorization for noise filtration in

binary images. The vector presentation was used for the construction of a reference raster image, which contains global context information about the input image. The context information was used for filtration of border noise in binary images. The main goal of the approach was to achieve a better compression of binary images by the elimination boundary noise. The vectorization procedure used algorithms and software presented in the publication **P1**. Due to the efficient implementation of the vectorization algorithm, the time cost of the vectorization phase is negligibly small (<10%) in comparison to the total processing time.

In paper **P2**, the author designed and implemented algorithms and software for the raster-to-vector conversion used in the feature-based filtering, and provided the vectorization experiments.

In the third paper (P3), a practical implementation of a Distance Transform (DT) based skeletonization algorithm on parallel multi-processor system is presented. The parallel realization of the algorithm is based on a procedure developed for the raster-to-vector conversion system presented in paper **P1**. The image is divided into blocks, and the blocks are distributed among the parallel processors. After processing of the data obtained with two passes over the block in two directions, the processors exchange the DT data on the borders. Then the processors perform a short additional run to complete the skeletonization process. The depth of the run is defined by the DT values on the borders of the blocks.

In paper **P3**, the ideas were developed and implemented by the author. He is the principal author of the paper. The second author contributed to the selection of the parallelization strategy.

In the forth paper (P4) we study the problem of optimal approximation of open N-vertex polygonal curve with minimum error for a given number of linear segments M (*min*-ε *problem*) with error measure L_2. The purpose of the algorithm is to fill the gap between fast but sub-optimal and optimal but slow algorithms. We introduce paradigm of *bounding corridor* in the state space, and *iterative reduced search* in the corridor. The paradigm is the used for constructing efficient algorithm for *min*-ε *problem* in question as well as for solving a number of other approximation problems, which are presented in publications **P5, P6, P7**.

The proposed iterative reduced-search algorithm consist of three steps: (a) find a reference solution with any fast heuristic algorithm; (b) construct a bounding corridor of fixed width W along the reference solution in the state space; and (c) perform search in the bounding corridor with Dynamic Programming (DP) method. We repeat the steps (b) and (c) of the procedure, using output solution as a reference

one for the next iteration. The time complexity of the developed algorithm is $O(W^2N^2/M)$, which is between $O(N)$ and $O(N^2)$, and the space complexity is $O(WN)$. Trade-off between performance and optimality can be controlled by width of the corridor and the number of iterations.

We also propose a modification of the state space in the case of *min*-ε *problem*, which reduces time complexity of the original full search algorithm in the case of large M. Then we offer a more efficient computational scheme for the algorithm, which reduces processing time by elimination of approximation error recalculation.

In the fifth paper (P5) we propose a fast near-optimal algorithm for solving the problem of *min*-# polygonal approximation of digitized curves. The algorithm is based on the reduced search approach introduced in **P4**. The algorithm consists of two steps. It first finds a reference approximation with minimum number of segments for a given error tolerance by using L_∞ error metric. It then improves the quality of the approximation by the reduced-search algorithm with additive L_2 error measure. To combine the practicality of the distortion measure L_∞ and the high visual quality obtained with the error measure L_2, we proposed to use the distortion measure with metrics L_∞ as an input control parameter ε, and the error with measure L_2 as the cost function for the optimization. The algorithm is tailored for high-quality vectorization of digitized curves. The time complexity of the algorithms varies between $O(N)$ and $O(N^2)$.

In the sixth paper (P6), we introduce a new approach for the *min*-ε and *min*-# approximation of *closed contours* based on dynamic programming method for open curves. It performs an approximation of the cyclically extended contour and then makes analysis of the state space to select the best starting point. The processing time is double of that of the approximation of the corresponding open curve. The time complexity of the algorithms is defined by the complexity of approximation algorithms for open curves in use.

In the case of *min*-# *problem*, the analysis of the state space is reduced to the analysis of one-dimensional array of parent states. In fact, if any closed approximation polygon of size $M/2$ or less starts and ends the same vertex, we can select this vertex as the optimal starting point. With the algorithms introduced for the approximation of closed contours, a solution can be found for double processing time compared to the case of open curve. For solving the *min*-ε *problem* the method suggested can be used along with the iterative reduced search algorithm with time complexity between $O(N)$ and $O(N^2)$.

In the seventh paper (P7), we consider the *multiple-object min-ε problem*, which can be formulated as an optimal approximation of K objects (curves) with a given total number of linear segments M. To find optimal solution of the problem in question, we have to find (1) the optimal distribution of segments number among the objects as well as (2) the optimal approximation of all the objects with the optimal number of segments.

At first, we introduce a general solution to the problem using a full search in the state space, and then we extend the iterative reduced search of **P4 t**o the case in question. The proposed algorithm includes three steps: a) approximation of the objects by reduced search in the *multiple-goal* bounding corridor, and calculation of the cost functions; b) calculation of the optimal allocation of the constrained resource among the objects using the calculated cost functions; c) restoration of the optimal solution for the found optimal number of segments. The procedure is then repeated iteratively several times (for near-optimal solution), or until no changes appear (for practically optimal solution). The time complexity of the proposed algorithm is between $O(N)$ and $O(N^2)$; and the space complexity is $O(N)$. Trade-off between performance and quality is controlled by the number of iterations and the corridor width. The developed algorithm is intended to be used for high quality vector data reduction.

In papers **P4-P7,** the author is responsible for developing the ideas, and implementation of the algorithms. He is also the principal author of the papers. The role of the second author has been mainly that of a supervisor.