Adaptive Image Segmentation for Managing Inventories of Museums

Robert Sablatnig Michael Dangl
Pattern Recognition and Image Processing Group
Vienna University of Technology
Favoritenstrasse 9/183/2, A-1040 Vienna, Austria
{sab,mll}@prip.tuwien.ac.at

Peter Stadler
Department of Prehistory
Museum of Natural History, Vienna
Burgring 7, A-1010 Wien, Austria
Peter.Stadler@univie.ac.at

1. INTRODUCTION

Museums try to archive their collection digitally since they should provide an easy access to data and objects that they possess in order to ease the answering of scientific questions and to fulfill the demands for examples of their collection. Furthermore, if photographs of their objects are published, they try to commercialize the image copyrights of these photos. So there is a strong need for the graphical and pictorial documentation of the objects possessed by a museum.

Figure 1: Typical archive photo.

Archives of museums contain thousands of photographs, which were used in the past 100 years to document the objects in storehouses as well as objects in exhibitions. To provide an access to this data and to save the pictorial information of old photographs, these images are stored in image databases. Usually, these databases are indexed by means of keywords that describe the content of the image. In the last few years, content-based image retrieval has become a very important field of research. The retrieval and selection of images from a collection is performed via features automatically extracted from the images themselves [5]. Such systems typically let users retrieve desired images from a collection on the basis of features representing color, texture, or shape - either singly or in combination [2].

Both of these databases (keyword-based or content-based) perform well, if the images stored in the database contain only one single object. Since photographs were costly, people tried to get as many objects as possible onto one photograph. Now these photographs are digitized to be stored in the database. To fulfill the condition one object per image, either the digitalization has to perform an object by object acquisition, or the photos are scanned once and the segmentation of the objects is performed afterwards. Figure 1 shows a typical archive photo containing archaeological finds. Some of these objects have a predefined catalog number which should also be included.

Currently, textual description and classification are generated along with the catalogs of images segmented manually. To save time and costs, the segmentation of objects should be automatic or at least semi-automatic. Therefore, we are developing an image segmentation and database client/server system for processing, generating and managing large, organized image collections of objects in museums. The system should provide a simple facility for inventory management, fast access to collections of finds, hierarchical catalogization of objects, and an easy-to-handle front-end to a web-server in order to provide local or world-wide access to the archive.

This paper shows a system that automatically segments images that contain multiple objects to be used as an input for the image database. Section 2 describes the system setup and the nature of the image data used for segmentation. Following the presentation of the actual segmentation technique in Section 3, the user interface and results are discussed in Section 4.

2. SYSTEM SETUP

The target system should provide an easy access to all objects and catalogs possessed by the museum. Therefore, all objects have to be digitized, most of them are already photographed in a standard way. Next, the digital images have to be segmented in order to represent each individual object and stored together with additional data like category, purchase date, and storage place. All data is stored in a database which is accessible via a restricted local or world-wide network (see Figure 2).

A scanner is the input device for photographs and illustrations producing images of 2000x2000 pixels in RGB-24bit color depth. Each image contains on average up to 60 objects that have to be segmented and categorized by archivists on “segmentation stations”. Afterwards the classification information and the image are stored within a database which provides interfaces for commonly used tasks like generation and update of web-catalogs.

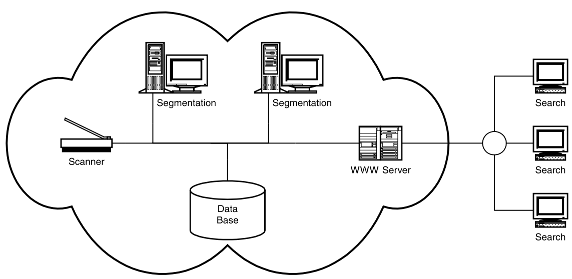

Figure 2: Setup of the working environment.

There are several requirements for the segmentation system:

- **Types**: Images contain photographs or line drawings.
- **Background**: The background is structured (paper with millimeter grid) or uniform.
- **Illumination**: The illumination is not uniform, some photos are taken using a combination of diffuse back-lighting [4] and diffuse front illumination [11], others were taken with diffuse front illumination only.

These assumptions and requirements allow us to form plausible hypotheses for an optimal segmentation but furthermore give raise to the possibility of adjusting segmentation parameters manually if the derived hypotheses fail for a certain image.

Initially, the following a-priori assumptions on the content of the images for the automatic segmentation are made:

- **Objects**: Objects to be segmented are darker than the background.
- **Overlap**: Objects do not overlap; if there are overlaps in the images these overlaps are supposed as overlaps of objects and tags mounted onto objects.

To improve efficiency, 2 levels of a 2x2/4 image pyramid [9], [10] of the blue color channel are built. This reduces the image size to 500x500 pixels, where the first step of image segmentation is performed [8]. If ambiguities in segmentation are detected, object borders are searched for at the original resolution.

3. IMAGE SEGMENTATION

The extraction of homogeneous regions in the image is called image segmentation. Homogeneity in the sense of region segmentation is defined as a connected set of pixel that share a common property such as intensity, color, texture or other local statistical indicators and are supposed to correspond to a physical object or surface. The regions should be simple and without many small holes, the boundaries of each segment should be non-ragged and spatially accurate, which is not always guaranteed in the actual case.

Wide-spread techniques have been developed for image segmentation, some considering general purposes and some designed for specific classes of images. A comparison between the different techniques is not easy, since, as stated in [6], no theory of image segmentation exists. Image segmentation techniques are basically ad hoc and differ precisely in the way they emphasize one or more of the desired properties and in the way they balance and compromise one desired property against another. However, there are three very common techniques: pixel classification, region growing and split-and-merge schemes. Further techniques like relaxation and spatial clustering schemes and more extensive treatment of these techniques can be found for example in [1], [17], [22] and [15].

Region growing starts with ”atomic” regions, which are individual pixel or small regions with uniform or nearly uniform pixel properties such as gray level, color, texture or alike. Neighboring regions are then merged based on their relative properties, like one region is largely including the other or the merged region has a more ”regular” shape [13]. Since region growing methods are usually independent of the kind of scene being observed, they are not sufficiently powerful to be relied upon to detect all the regions expected and no more [19].

Split and merge methods use both region splitting (like thresholding) and region merging (like region growing) to produce a segmentation of the image. Since this method is also using region growing methods, similar drawbacks occur. The main difference is the iterative refinement of the segmentation, and that the atomic cells for the region growing are provided by the splitting technique, which is introducing problems to the segmentation resulting in block structured segmentation [6].

The simplest approach to segment an image is to classify each pixel based on some image property like intensity or color. The property like intensity is divided into intervals and every pixel with a certain intensity lying within a particular interval is assigned to this class. Connected regions in this class provide the desired segmentation. This method is suitable for scenes containing a homogeneous object with a high contrast to a uniform background. The crucial question of this method is how to choose the thresholds. Typically they are selected by experience or from other a priori knowledge (see [21] for a survey of threshold selection techniques). One possibility is to take advantage of the multi-modal nature of typical gray-level histograms. If the histogram has clear peaks, these peaks give the threshold for the corresponding region [16].

The simplicity of this method has its drawbacks since pixel assigned to a single class are not necessarily members of coherent regions as the spatial positions of the pixel are not taken into consideration during segmentation. The threshold selection is not unique if there are a plenty of local maxima in the histogram. However, there are several techniques that try to overcome these difficulties like the super-spike method, that tries to smooth the histogram to give pronounced peaks [12] or a segmentation refinement method by [14] that imposes the spatial coherence on the segmentation by using the Markov-Random-Field model [3].

Due to their simplicity and low consummation of computing time [20] our approach to image segmentation is also based on pixel classification combined with region growing and split and merge basics. Figure 3 shows the workflow of the implemented segmentation algorithm.

Figure 3: Workflow of the segmentation process.

Since the quality of the scans is varying due to quality differences of the original photographs, a histogram equalization [7] is applied to improve the contrast of the images containing multiple objects. Then 2 levels of the gray-level image pyramid are calculated and the top level is taken for further computations (multi-scale representation in Figure 3). Afterwards, a background initialization takes place based on the assumption that pixels on the image border are background pixels. These pixels are used to perform region growing in order to initialize the background. Next, the image is split into 50x50 pixels windows, the local histogram is computed for each window and the threshold is determined adaptively within each window to segment the objects from the background (adaptive thresholding in Figure 3). The complete image is constructed out of the segmented windows forming a binary mask image.

In the lower resolution, adjacent objects tend to form one object, if the distance between them is relatively small (7- 8 pixel) in the original resolution. Therefore, a morphological closing [18] applied to the binary top level image is used to separate them from one another. Of course this only works if the original distance of almost touching objects is above 7 pixel.

The binary image of the top pyramid level is taken to label every detected object. The process uses a scan-line technique, every detected object pixel is taken into one region as long as there is a neighboring object pixel (8-neighborhood). This generates a mask image where every pixel that belongs to the same object gets the same label. Since there are ambiguities (2 objects fall into the same class since their distance is smaller than 7 pixel in the lowest level) every segmented object is checked in the original resolution. The result is the segmented image, displaying the labeled objects in color.

4. USER INTERFACE AND RESULTS

Figure 4 shows the main window of the segmentation system along with the identified objects of the image in Figure 1 and, on the right side, the object thesaurus. After processing the automatic segmentation, each object which was identified by the system is presented to the user for an individual check. If the automatic segmentation process fails to calculate the correct area it is possible for the user to adjust segmentation parameters manually, either for the whole image or for a selected region within the image. Furthermore, the user has the possibility to separate or merge image regions manually by drawing separation lines or selecting two regions to merge.

For faster execution, the toolbar of the application represents the typical workflow of the archivation process: load an image file, start automatic segmentation, eventually adjust segmentation parameters manually, split or merge detected image regions and finally store the gathered information in the database system.

The object thesaurus is the module which handles user input for object parameters and communicates with the database system. For each category of finds it is possible to design a template. This template comprises all object parameters which are necessary for a human expert to classify an archeological find.

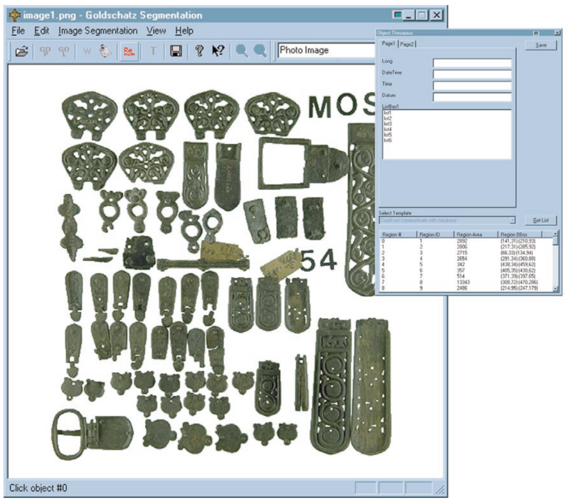

Figure 4: User interface for segmentation.

Segmentation results for a test series of 20 images consisting of both photographs and line drawings show a success rate of 99% of correctly determined image objects. Out of 569 image objects, 564 were found with correct object boundaries of which 11 are objects which are overlapped by one or two pixels with another object. Four objects were split up into two objects each since they show narrow object boundaries (thin objects). One object was not found by the application because the contrast to the background pattern was insufficient. The average computation time on a Pentium II-350 with 128 MB RAM is 2,27 sec. for an image with 2000 by 2000 RGB pixels.

5. REFERENCES

[1] D.H. Ballard and C.M. Brown. *Computer Vision.* Prentice Hall, 1982.

[2] T. Catarci, M.F. Costabile, S. Levialdi, and C. Batini. Visual query systems for databases: A survey. *Journal of Visual Languages and Computing*, 8(2):215–260, 1997.

[3] G.R. Cross and A.K. Jain. Markov random field texture models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 5(1):25–39, 1983.

[4] E. Downing and K. Doyle. Test and inspection II: Lights, camera, inspection. *Industrial Process Control Magazine*, 59(6):71–80, 1986.

[5] V.N. Gudivada and V.V. Raghavan. Content-based image retrieval-systems. *Computer*, 28(9):18–22, September 1995.

[6] R.M. Haralick and L.G. Shapiro. Image segmentation techniques. *Computer Vision, Graphics and Image Processing: Image Understanding*, 29:100–132, 1985.

[7] R.M. Haralick and L.G. Shapiro. Glossary of computer vision terms. *Pattern Recognition*, 24:69–93, 1991.

[8] T. Jiang, M.B. Merickel, and E.A. Parrish, Jr. Automated threshold detection using a pyramid structure. In *Proc. of 9th. Intl. Conference on Pattern Recognition, Rome, Italy*, pages 689–692, 1988.

[9] J.M. Jolion and A. Rosenfeld. *A Pyramid Framework for Early Vision: Multiresolution Computer Vision.* Kluwer, 1994.

[10] W.G. Kropatsch and S. Ben-Yacoub. A revision of pyramid segmentation. In *Proc. of 13th. Intl. Conference on Pattern Recognition, Vienna, Austria*, volume B, pages 477–481, 1996.

[11] S. Misra and C.A. Bennett. Lighting for visual inspection task. *Light. Des. Appl.*, 13(1):32–33, 1983.

[12] K.A. Narayanan and A. Rosenfeld. Image smoothing by local use of global information. *IEEE Trans. Systems, Man and Cybernetics*, 11(12):826–831, December 1981.

[13] R. Nevatia. Image segmentation. In T.Y. Young and K.S. Fu, editors, *Handbook of Pattern Recognition and Image Processing*, pages 215–231. Academic Press, 1986.

[14] T.N. Pappas and N.S. Jayant. An adaptive clustering algorithm for image segmentation. In *Proc. of Intl. Conference on Computer Vision, Tampa*, pages 310–315, 1988.

[15] T. Pavlidis. *Structural Pattern Recognition.* Springer Verlag, Berlin, 2 edition, 1980.

[16] J.M.S. Prewitt. Object enhancement and extraction. In B.S. Lipkin and A. Rosenfeld, editors, *Picture Processing and Psychopictoris*, pages 75–149. Academic Press, New York, 1970.

[17] A. Rosenfeld and A.C. Kak. *Digital Picture Processing.* Academic Press, New York, 1982.

[18] J. Serra. Morphological filtering: An overview. *Signal Processing*, 38(1):3–11, 1994.

[19] Y. Shirai. *Three-Dimensional Computer Vision.* Springer Verlag, Berlin, Heidelberg, New York, 1987.

[20] C. Torras. *Computer Vision, Theory and Industrial Applications*, chapter Segmentation, pages 59–95. Springer Verlag, New York, 1992.

[21] J.S. Weszka. A survey of threshold selection techniques. *Computer Graphics Image Processing*, 7(2):259–265, April 1978.

[22] S.W. Zucker. Region growing: Childhood and adolescence. *Computer Graphics Image Processing*, 5:382–399, 1976.