32

Shape Information in an artefact Database

K.J. Goodson*

32.1 Introduction

Advancing fields of computing hardware technology have provided the possibility for acquisition, storage and display of large amounts of pictorial and geometric information. As a result, much work has been carried out on the creation and use of image databases (Iyengar & Kashyap 1988). The demands which this type of work place on the currently available software technology are great. Unless explicit information concerning a picture's content is available then the image content of a database is simply a catalogue for visual interpretation. For these data to have greater meaning, the ability to acquire, store and interrogate explicit shape information, and higher level forms of data abstraction concerning the elements within an image, is required.

Until research into computer vision provides robust methods for automated image understanding of complex scenes, and until database architectures are developed to provide spatial data types and languages for spatial enquiries(Orenstein & Manola 1988), the scope for moderately intelligent image databases is limited. The available options are to manually supply key feature information or to constrain the images contained in the database to those which lend themselves to automated object description. Custom applications can then be built for dealing with geometric and spatial enquiries. This paper describes the aspects concerning shape information, from acquisition through to interrogation in a project to develop a *Graphically Oriented Archaeological Database (GOAD)*, (Goodson 1988). Fig. 32.1 shows the state of the computer screen during a consultation with an image database. The user interacts by directing input from the keyboard and mouse into various windows, text fields, push buttons, switches, bit map images and pop up menus by positioning the arrow shaped cursor over the appropriate screen object.

32.2 Shape acquisition

In creating shape information from two-dimensional bit map images the important aspects are to achieve sufficient resolution, accuracy and semantic correctness of the shape feature codes. This tends to preclude entirely manual methods of shape acquisition, mainly because the required sampling interval for this first generation shape description must be high, particularly in small areas that provide good discriminatory features.

The central resource of images for acquisition is the large set of existing line drawings, originally produced for conventional cataloging (see Figs. 32.1 and 32.4a). The reasons for choosing this type of image rather than, for example, photographs of real artefacts are numerous:

- their familiar nature to an archaeologist who hopefully will find his or her computer screen as comfortable to work with as a real desk top;

* Department of Electronics & Computer Science
University
Southampton S09 5NH

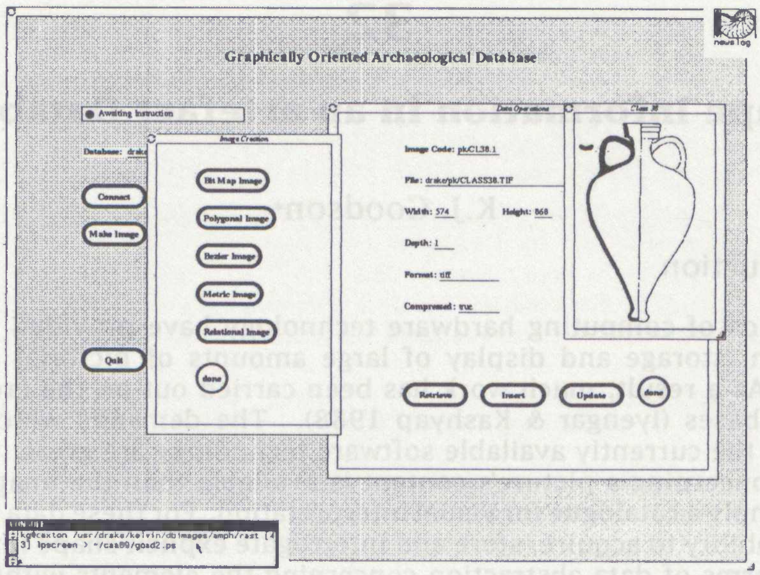


Figure 32.1: A Consultation with a Graphical Database Application

- their suitability for use with robust feature extraction algorithms;
- the increased portability due to the possibility for display on a wide range of hardware;
- the ease of data compression of the raw digital image matrices.

Digital images are acquired using either a video camera and video frame grabber, or by using a document scanner. Whilst images of line drawings obtained with a document scanner are usually superior to those captured with a video camera, both methods are pursued to provide the flexibility image input by end users. The characteristics of the images obtained by these two methods are sufficiently different to require different forms

of processing to achieve object descriptions. Figure 32.5 shows a snapshot of the computer screen whilst running an image processing application program. This experimental environment has been created to test various image processing algorithms on the artefact images.

32.2.1 Acquisition of shape primitives from video images

A technique which has proved successful for the initial image processing to provide shape primitives from images of line drawings acquired with a video camera is based on (Watson *et al.* 1984) and is described in (Goodson & Lewis 1988). A brief overview of the process follows.

An idealised *top hat* intensity profile is illustrated in Fig. 32.2. When this type of profile is convolved with a gaussian operator the result is a ridge–like profile. A property of such a profile is that the average intensity of points adjacent to some central point is less than the intensity of the central point. Therefore, for a two-dimensional digital image the logic operation shown in Fig. 32.3 can be applied to distinguish between points that are on ridge type profiles and those that are not. This operation simply states that:

> **if** the average intensity of the neighbours of a central pixel in a 3×3 pixel region, multiplied by some constant slightly greater than one (e.g. 1.05) is

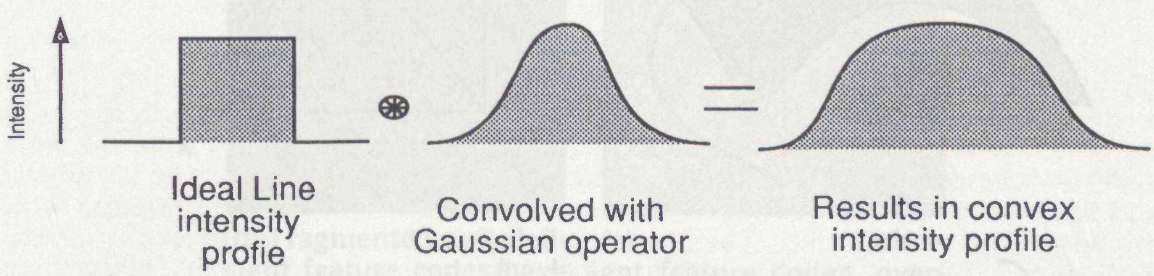
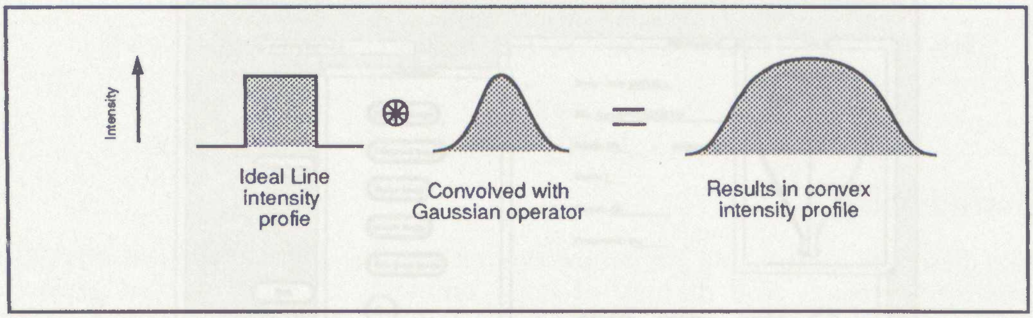
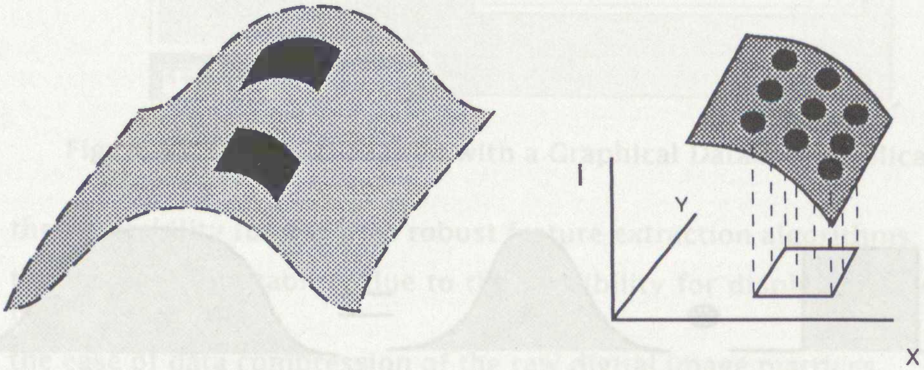


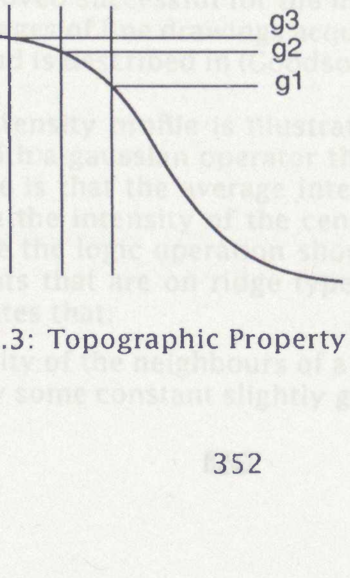
Figure 32.2: Intensity Profiles





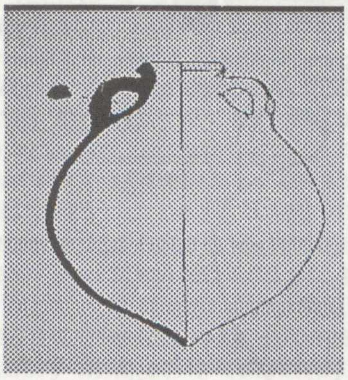
$$g'(x,y) = \begin{cases} g(x,y) & \text{if } g(x,y) * C >= \left(\sum_{y'=-1}^{1} \sum_{x'=-1}^{1} g(x+x',y+y') - g(x,y) \right) / 8 \\ 0 & \text{if } g(x,y) * C < \left(\sum_{y'=-1}^{1} \sum_{x'=-1}^{1} g(x+x',y+y') - g(x,y) \right) / 8 \end{cases}$$

$$(C = 1 + \delta)$$

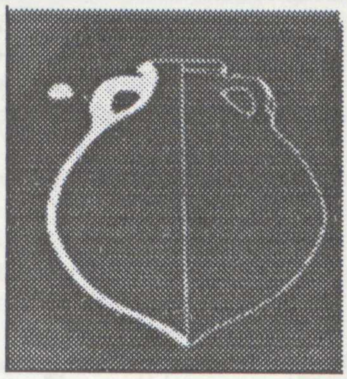


$$g2 > (g1 + g3) / 2$$

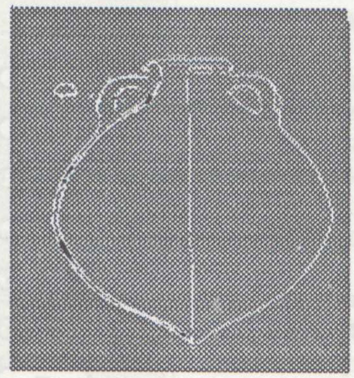
Figure 32.3: Topographic Property of Convex Profiles



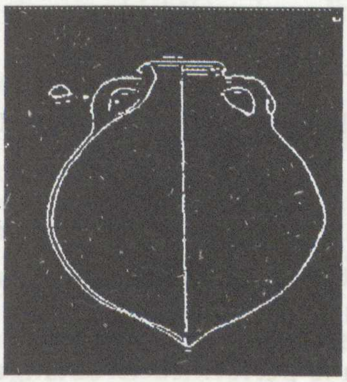
(a) A Raw 256 × 256 Image matrix captured with a video camera



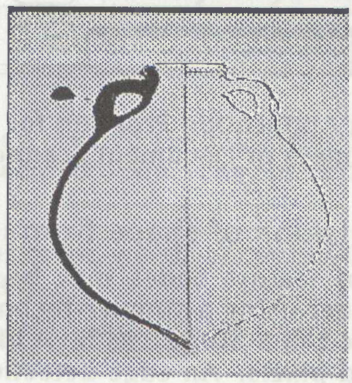
(b) The image has been negated and blurred



(c) Convex topographic profiles have been extracted



(d) Fragmented and deficient feature codes have been extracted



(e) Coherent feature codes, overlaid on the original in white, generated using a semi-automated feature extraction system

Figure 32.4: Sequence of operations in tracking lines from a video image

less than the intensity of the central pixel
then retain the current central pixel's intensity value
otherwise set the central pixel's intensity value to zero.

This results in an image containing background at zero grey level, and ridge profiles retaining their original grey level profiles (see Fig. 32.4c). The output image matrix can be used as input to a tracking operation which scans the image in raster mode until a pixel which exceeds a dynamically set threshold is encountered. The best set of starting pixels for a line in this region are found and line tracking proceeds by following the ridge top to its end. A vector code for each pixel step along the line is recorded. Scanning for further lines then resumes at the start of the tracked line segment. Deficiencies in the thresholding processes and characteristics of the original image result in fragmentation of the image features and creation of spurious features (see Fig. 32.4d).

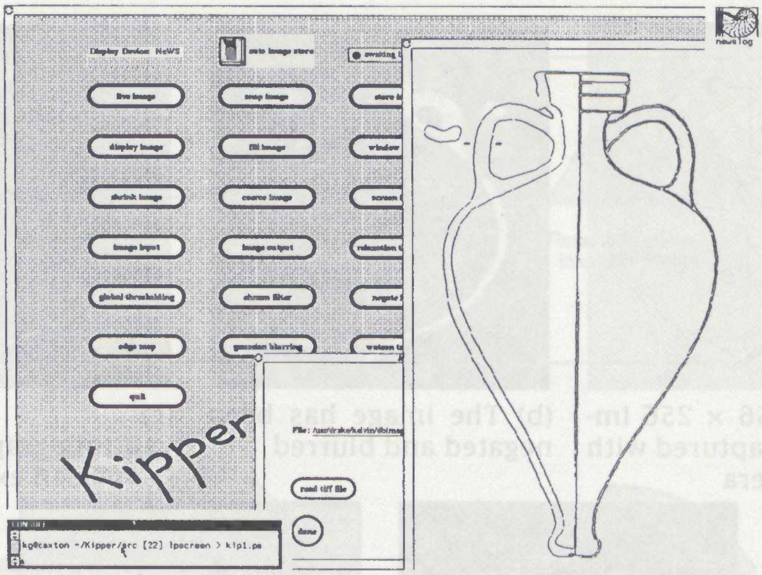


Figure 32.5: An Application Program for Obtaining and Processing Images captured with a Video Camera or Document Scanner

32.2.2 Acquisition of shape from images from the document scanner

A similar set of feature codes may be obtained from an image acquired with a document scanner using a simpler technique. This technique draws from the work of Ali and Burge (Ali & Burge 1988). A binary image is considered to contain object and background pixels. An edge detection is performed to obtain a one pixel wide boundary map of the object. The disconnected edge pixels are now classified on the basis of two criteria; the number of edge pixels in the eight connected neighbours and the number of edge to background and background to edge transitions encountered when following a path around the eight connected neighbours of the edge pixel. The classification is as follows :-

transitions	neighbours	classification
0	0	Isolated point
2	≤ 2	End point
4	2	Internal point
>6	-	Attractant Multi-way junction
4	>2	Potential Multi-Way junction

Groups of potential multi-way junctions are clustered into a single attractant multi-way junction. Tracking proceeds by searching for end points, and then following internal points until an attractant multi-way is encountered. After all open ended lines have been tracked looped lines are found by tracking from the first untracked internal points.

32.2.3 Overcoming processing deficiencies

Neither of the schemes for automatically extracting shape information are sufficiently robust to provide reliable feature codes. For video images it has been found that manual intervention after line tracking is the most suitable method for overcoming the deficiencies of the tracking processes. A scheme whereby feature fragments may be assembled into coherent labeled features has been developed and is described in (Lewis & Goodson 1988). The system developed is a *Semi-Automated Feature Extraction, (SAFE),* system. Using SAFE, complete features are built up by sequentially joining feature fragments. The state of the feature is displayed graphically as an overlay on the original image. SAFE automatically joins fragments to make features until ambiguities are encountered, at which point the user is asked to intervene to resolve the ambiguity by accepting or rejecting members of a set of candidate feature fragments, or by selecting an alternative fragment using a cross-wire cursor. Mistakes can be rectified and segments added using a set of feature editing facilities. Knowledge based techniques for reducing the amount of manual intervention have been investigated and are presented in (Lewis & Goodson 1988).

With document scanned images, however, it is possible to intervene manually before the tracking process. A bit map editor for cleaning and manipulating the binary images is currently being developed. The ability to erase unwanted object areas, and to complete small breaks in a boundary, followed by flood filling of interior regions should allow line feature extraction processes to be applied with confidence that the features obtained are coherent and meaningful. The editor is destined to become part of the graphical database front end.

32.3 Shape representation

Once a primitive shape description has been created the organisation and representation of the stored description must be decided upon. The types of representation could be one or a number of the following :-

- reversible transformations (e.g. fourier or walsh encoding (Chellapa & Bagdazian 1984, Sarvarayudu & Sethi 1983)),
- codings made relative to some fidelity criteria (e.g. polygonal approximation, curve fitting) (Wu 1984),
- wholly irreversible global measures (moments, euler number, minimum enclosing rectangle (Gonzales & Wintz 1987)),
- or relational descriptions of the image contents (e.g. facts expressed in prolog (Ballard & Brown 1982)).

The relative merits of these various forms of representation are currently being examined. A number of these representations are used particularly for their shape matching properties, whilst techniques such as curve fitting provide aesthetically pleasing visual representations but cloud the issue of shape matching. Fig. 32.6 illustrates the essence of a number of these techniques. However shape matching might be attempted, normalisation techniques will usually need to be employed to account for differences in scale, orientation, translation and the starting position of the boundary trace. Matching in the spatial domain is usually achieved by using a distance function such as least squares to compare a reference outline with some sample outline. Whilst in the frequency domain, an estimate of similarity is given by the correlation of the two spectral functions (Ballard & Brown 1982). Setting aside the type of representation for the shape, the information must be organised in a fashion best suited for ensuring that matching one shape against another is meaningful. This can be achieved by supplying tags on the shape, naming various corresponding parts. A sufficiently flexible approach to the format in which a shape is described is necessary, allowing multiple representations of the shape in the most suitable schemes. For example, in the case of complete amphorae, two shape outlines are stored, one of the full outline, and one without the handles. If multiple representations are present then a matching scheme must have the ability to collect

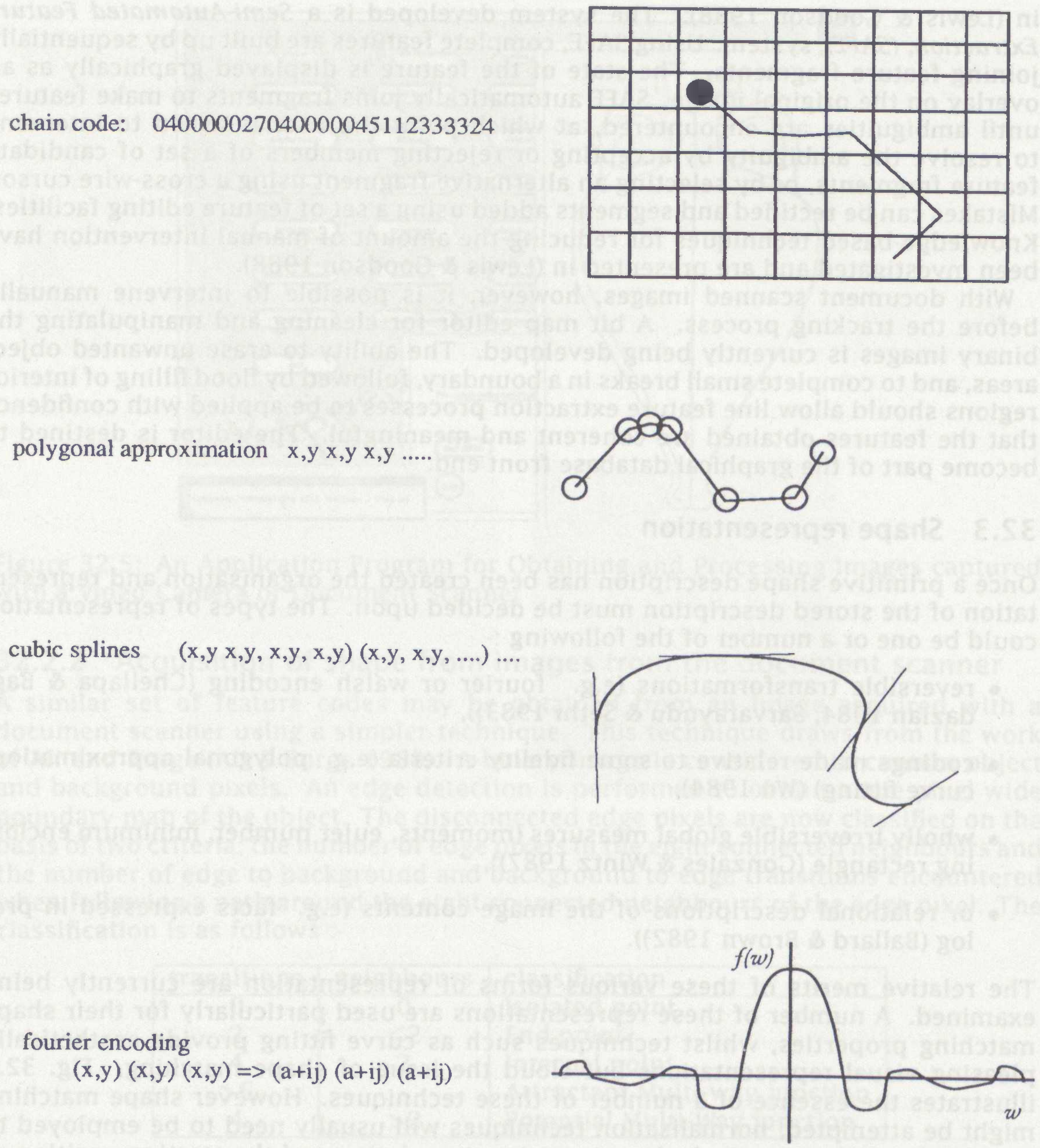


Figure 32.6: Shape Representation Schemes

evidence from the various heuristics. This suggests the use of artificial intelligence techniques in evaluating the certainty and is an area for further investigation.

32.4 Shape storage

The large irregular data structures needed to hold shape information conflict with the highly structured requirements of many database architectures. The choices for storage of shape information are therefore:

- to choose a representation which is highly structured, in order that it can be held within the database;
- to hold data outside the database and keep pointers to that data from within the database;
- to select or develop a database architecture which is sufficiently flexible to be capable of storing irregular data structures.

Initial frameworks for storage and display of images and shapes have relied on holding their representations outside of the database system. A file name is stored in the database along with attributes of the file and the image. The images and shapes are accessed and displayed using a set of routines written in C which interface with the database management system and the NeWS PostScript graphics server (see Fig. 32.1).

An alternative solution would be to create a transparent user interface to a dual database system which retains the relational database management system for storage of alphanumeric data, whilst using an object oriented (Roussopoulos *et al.* 1988, Mohan & Kashyap 1988) database model for storage of images (see Fig. 32.7). The structure of the interface imposes constraints in order that parsing of requests into the separate parts for each database management system is possible. A formulation of an example query is shown in Fig. 32.8.

The class hierarchy of the object oriented database management system contains the class 'image', which is the root node of a hierarchy of subclasses corresponding to various levels of information abstraction (Fig. 32.9). Methods for each subclass can allow an instance of a class to do simple things such as display itself, as well as more complex tasks such as shape matching or discriminant analysis in order to perform similarity analysis between images.

32.5 Conclusion

The potential for archiving and publishing large amounts of pictorial information, along with conventional alphanumeric data is large. However, the associated problems in selecting and presenting appropriate information in a flexible manner to a consumer of this information are equally large. By constraining the pictorial content, automated approaches to image understanding can, to a certain extent, provide key information for image retrieval based on shape similarity measurements. The use of alternative database structures, such as an object oriented scheme, would provide a better framework for storing irregular structures, aggregating large structures of similar type, and providing type specific methods for operating on these structures, thus providing the desired flexible approach to the organisation of shape matching.

Bibliography

ALI, S. M. & R E BURGE 1988. "A New Algorithm for Extractiong the Interior of Bounded Regions Based on Chain Coding", *Computer Vision, Graphics and Image Processing*, 43 (2): 256–264.

BALLARD, D. & C. BROWN 1982. *Computer Vision.* Wiley.

CHELLAPA, R. & R BAGDAZIAN 1984. "Fourier coding of Image Boundaries", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6: 102–105.

GONZALES, R. C. & P. WINTZ 1987. *Digital Image Processing.* Addison–Wesley.

GOODSON, K. J. 1988. "A Graphically Oriented Archaeological Database", *Archaeological Computing Newsletter*, 14: 16–20.

GOODSON, K. J. & P. H. LEWIS 1988. "Feature Extraction from Line Drawing Images", *in* Kitler, J., (ed.), *Proceedings of the Fourth International Conference on Pattern Recognition*, Lecture notes in Computer Science 301, Cambridge, U. K. Springer Verlag.

IYENGAR, S. S. & R L KASHYAP 1988. "Guest Editors Introduction: Image Databases", *I. E. E. E Transactions on Software Engineering*, 14 (5): 608–610. special section on image databases, pp. 611–688.

LEWIS, P. H. & K. J. GOODSON 1988. "Knowledge based interpretation of line drawing images", *in Proceedings of the International Association for Pattern Recognition's Workshop on Computer Vision*, pp. 216–219, Tokyo.

MOHAN, L. & R L KASHYAP 1988. "An Object Oriented Knowledge Representation for Spatial Information", *IEEE Transactions on Software Engineering*, 14 (5): 675–680.

ORENSTEIN, J. A. & F A MANOLA 1988. "PROBE Spatial Data Modelling and Query Processing in an Image Database Application", *IEEE Transactions on Software Engineering*, 14 (5): 611–629.

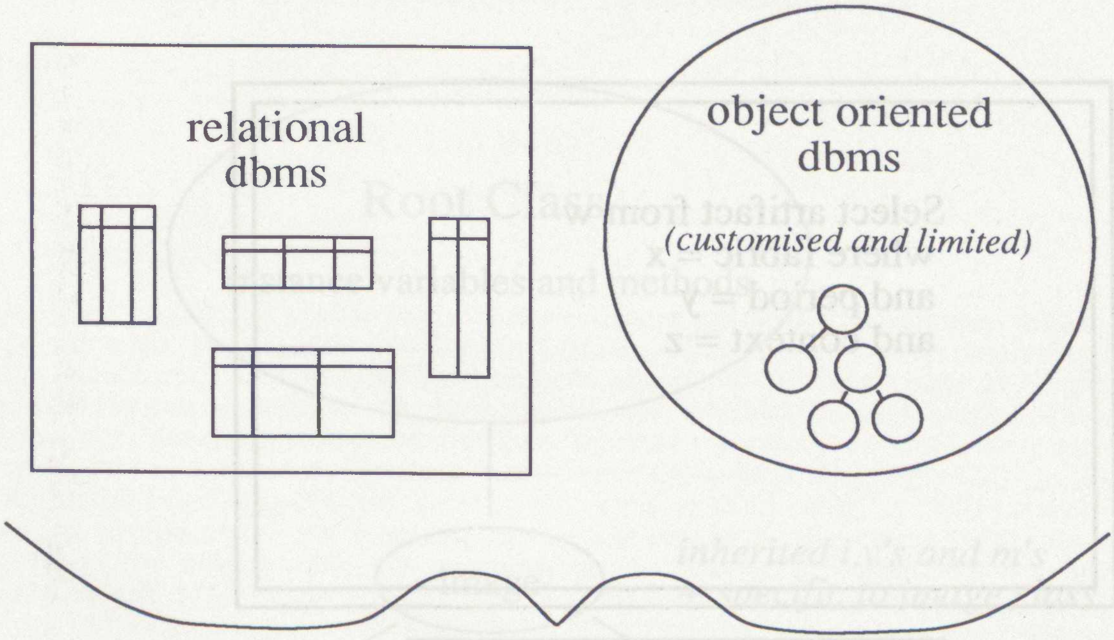
ROUSSOPOULOS, N., C FALOUTSOS, & T SELLIS 1988. "An Efficient Pictorial Database System for PSQL", *IEEE Transactions on Software Engineering*, 14 (5): 639–650.

SARVARAYUDU, G. P. R. & I. K. SETHI 1983. "Walsh Descriptors for Polygonal Curves", *Pattern Recognition*, 16 (3): 327–336.

WATSON, L. T., K. ARVIND, R. W. EHRICH, & R. M. HARALICK 1984. "Extraction of Lines and Regions from Grey Tone Line Drawing Images", *Pattern Recognition*, 17 (5): 493–507.

WU, L. D. 1984. "A Piecewise linear approximation based on a statistical model", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6: 41–45.

Hybrid System



transparent user interface

RDBMS
- stores conventional alphanumeric data
- performs joins etc.

OODBMS
- supports class hierarchy
- methods responsible for display etc.

Figure 32.7: A Hybrid Database management system

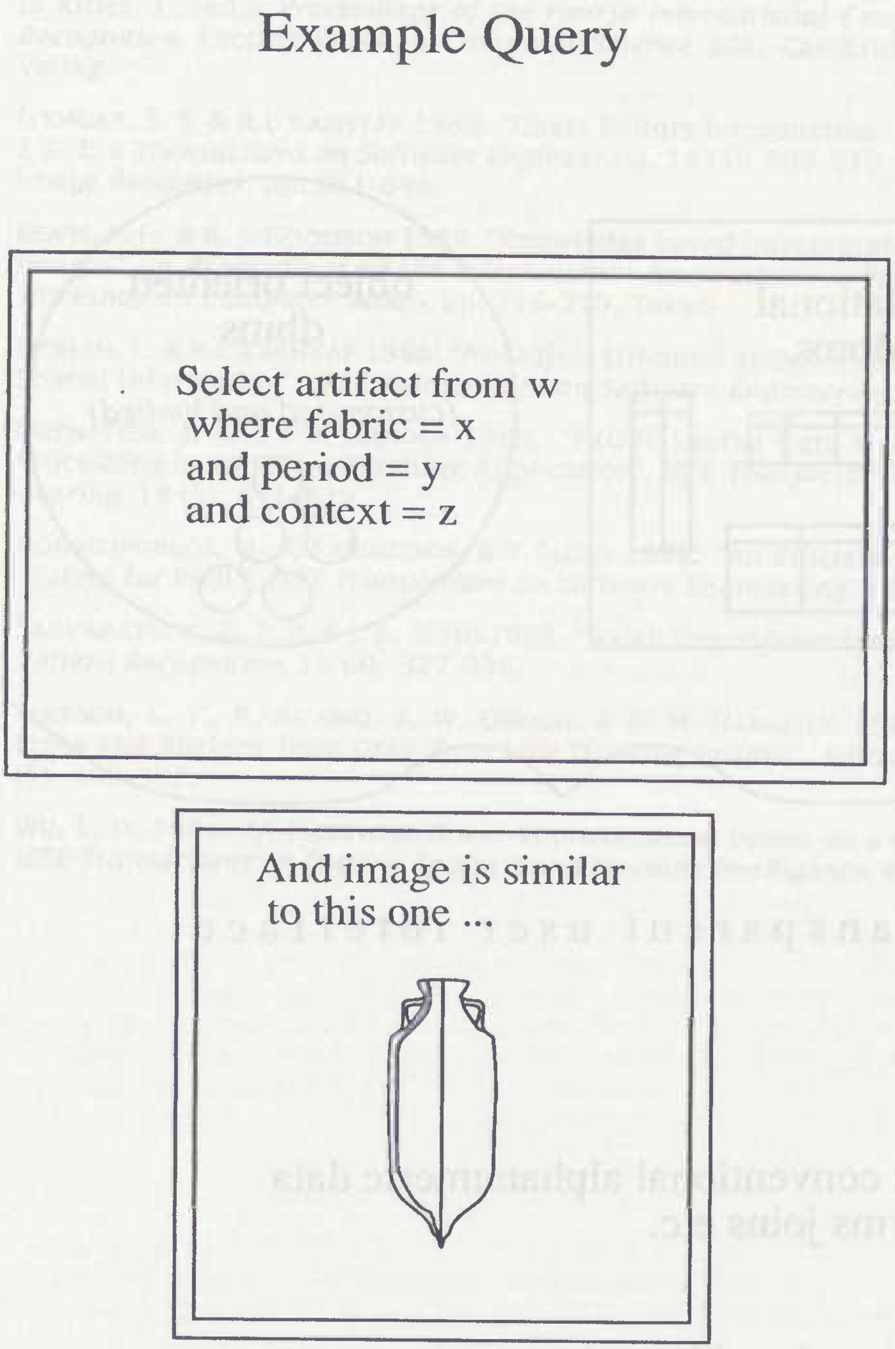


Figure 32.8: An example query in the proposed system

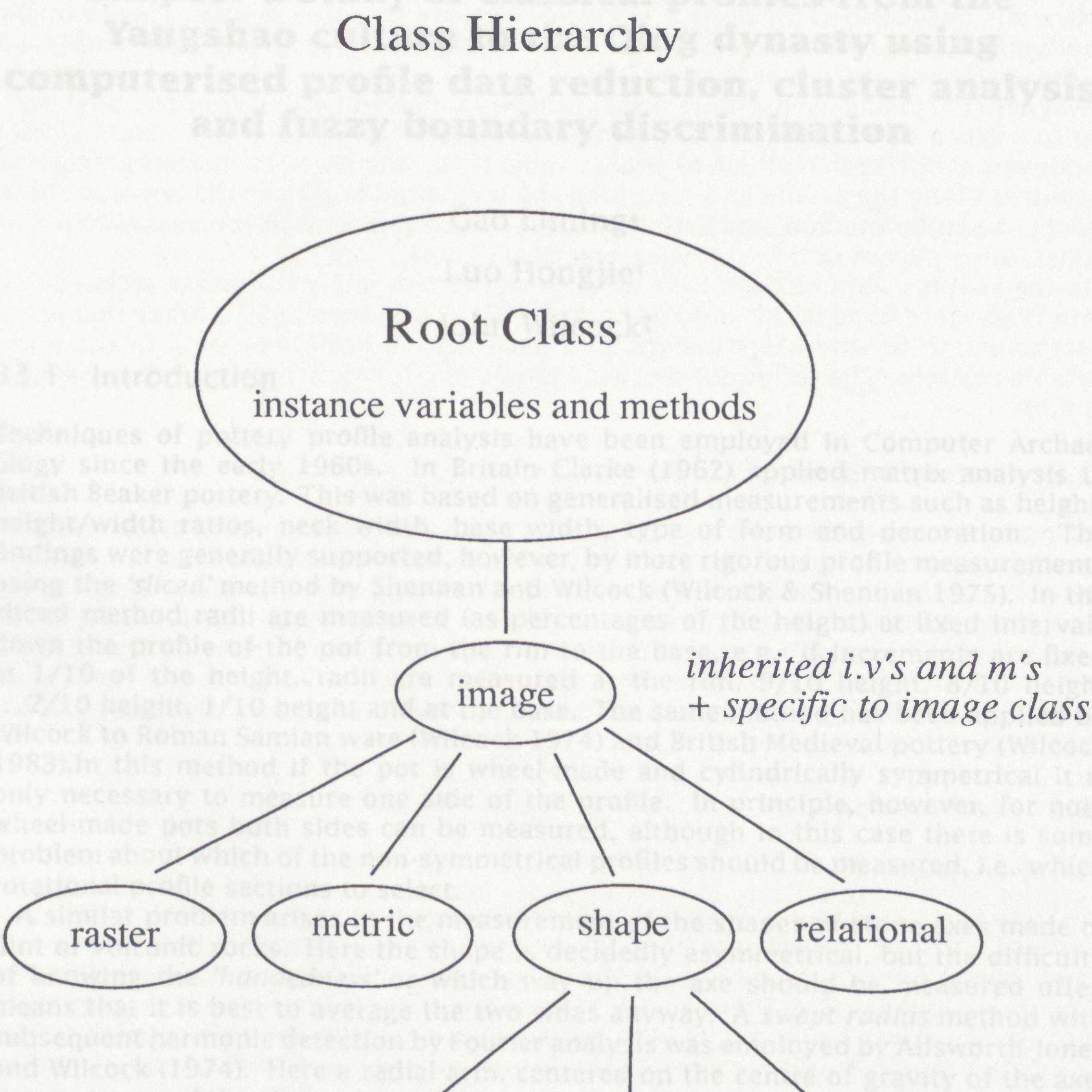


Figure 32.9: A Possible Object Oriented Database Class Structure