Vision Systems Laboratory

Book Chapter

Draft #1
Monday, 12 March 2001

Vision Systems Laboratory, School of Electronic Engineering,
Dublin City University, Dublin 9, Ireland.

Tel: +353 1 700 5489 Fax: +353 1 700 5508 Email: paul.whelan@eeng.dcu.ie

Chapter Z

Automated Cutting Of Natural Products: A Practical Packing Strategy

P.F.Whelan[1]

“"No program can deal flexibly with components of arbitrary shapes or with unanticipated failures that a human being would easily detect. It seems wiser, then, to abandon the goal of human flexibility and seek engineering solutions consistent with the limited capacities of present-generation robots. Better methods of standardization can obviate the need for human flexibility, and they have the advantage of working."
(Dreyfus and Dreyfus 1986 [1])

Z.1 Introduction

This chapter is concerned with the issues involved in the automated packing of two-dimensional objects into a two-dimensional bounded region, where the size and shape of both the objects and the region in which they are to be packed are not known prior to packing. The packing of such regions is directly related to the stock cutting task, one in which we try to maximise the amount of items that can be extracted for a given material (e.g. how are leather templates arranged on an animal hide so as to cut the hide efficiently?). Problems relating to the automated packing and nesting of irregular shapes are not only of theoretical importance, but have considerable industrial interest.

The automated cutting and packing strategy outlined in this chapter consists of two main components. The first provides a means of manipulating the shape and scene image at a geometric level. The second component consists of a rule based geometric reasoning unit capable of deciding the ordering and orientation of the shapes to be packed. The heuristic component must be capable of

[1] Director - Vision Systems Laboratory, School of Electronic Engineering, Dublin City University, Ireland. paul.whelan@eeng.dcu.ie, www.eeng.dcu.ie/~whelanp/home.html

dealing with the system issues arising from a specific application demand. This task can be simplified by maximizing the use of the information available from the product, process and the environment for a specific industrial application. The use of heuristic methods increases the generality of the packing system, thus making the development of procedures for new applications less cumbersome. One of the key features of this system is that it works towards an efficient solution, accepting that we cannot guarantee reaching an optimal solution. Therefore a mechanism for quantifying the packing systems performance will be necessary. This will enable a quantitative comparison of packing procedures [2,3].

Z.2 The Packing/Cutting Problem

Early research into determining optimal packing/nesting configurations can be traced back to Johannes Kepler in 1611 when he tried to determine if the most efficient method of packing identical spheres was an arrangement now known as face-centred cubic lattice. This consists of placing a bottom layer of spheres in a bounded region. Each successive layer is then arranged so that the spheres occupy the gaps of the layer below [4,5] (i.e. the arrangement greengrocers commonly use to stack oranges). Although the stacking of items such as oranges in this way seems intuitive, researchers are still unable to prove that this stacking configuration is the most efficient (Hinrichsen, Feder and Jossang [6] have concentrated on a simplified version of this problem and have developed strategies for the random packing of disks in two dimensions).

Research has shown that an optimal algorithmic solution for even the simplest, well-defined packing problem, such as pallet packing, is unlikely [7]. However, the aim of the system described in this chapter is more ambitious and complex. It's objective is to allow the flexible packing of random two-dimensional shapes into previously undefined scenes (the term *scene* is used when referring to a region of space into which we are required to place an arbitrary *shape*). The NP-complete nature of the simpler packing problem has a major bearing on the line of research taken. Any attempt at developing an optimal solution to the packing of random shapes, even if it did exist[2], would be difficult. It would also be difficult to constrain the problem, especially considering that it must deal with unpredictable shapes. Hence, the aim has been to produce an *efficient* packing strategy that is flexible enough for industrial use. To achieve this objective, the systems approach to the packing problem is essential.

Other approaches to the packing problem include, single pattern techniques such as dynamic programming, and multi pattern strategies such as linear programming [8,9]. Unfortunately these techniques do not meet all the requirements of a flexible packing system. In the former case, the aim is the generation of an optimal solution, and as such the approach would not seem to hold promise. The second technique, while useful for one-dimensional packing

---

[2] It is impossible to guarantee that an optimal procedure for the more general packing problem can be found, especially when you consider the NP-complete nature of the simpler pallet-packing task.

applications, is difficult to implement and use in two-dimensions. Alternatively, a heuristic approach to packing can be adopted. This line of research would seem to be the way forward if the system is to remain flexible and have the ability to cope with random shapes.

Another key element in the development of automated packing systems, concerns the method of shape description. The majority of current systems rely on correlation methods and low level features such as curve [10,11] and/or critical point matching techniques [12]. Other techniques enclose the shape of interest within a bounding rectangle [13], polygon [14] or convex hull, prior to paving the region to be packed with these predefined shapes.

Z.2.1 The One-dimensional Packing Problem.

Initial work on the development of automated packing systems concentrated on one-dimensional packing [15]. This can be best explained by considering a group of holes of similar size and cross-section, and a selection of boxes of varying length, but which have the same cross-section as the holes. The one-dimensional packing problem consists of placing all the boxes into the holes without any protruding. This area of research is more commonly referred to as *bin packing* [16], and it is one of the "celebrated problems of computer science" according to Hofri [17]. Sample one-dimensional packing applications include process scheduling (industrial and computer level), timetabling and the efficient packing of advertisements into a time slot between programs. Chandra, Hirschberg and Wong [18] relate the problems involved in the design of distributed computer systems, such as processor allocation and file distribution, to bin packing.

Z.2.2 The Two-dimensional Packing Problem.

One-dimensional packing problems can be extended to two and three dimensions. For example, the objective of the simplest two-dimensional task is the packing of a number of flat rectangular pieces into a rectangular scene. In the case of two-dimensional depletion, the task is the division of a large rectangle into smaller ones. Applications of two-dimensional stock cutting include the cutting up of materials, such as sheet metal [19], fabrics, wooden planks, glass and rolls of paper, into smaller pieces. The aim of such systems is to minimize the amount of waste material, referred to as trim loss, produced by the cutting process. One of the main applications for these techniques is in the area of cloth and leather cutting [20,21]. The minimization of waste material is especially important in the leather industry, for example shoe making [22], since the waste leather material cannot be recycled.

Z.2.3 The Three-dimensional Packing Problem.

In the three-dimensional case the objective is to pack rectangular blocks into a large empty space such as a rectangular container. This is usually referred to as the *knapsack problem*[3] due to its original formulation as a problem concerned with packing as many items into a knapsack before a hike. The three-dimensional depletion task consists of segmenting a rectangular box into a number of smaller rectangular boxes.

Z.3 Review of Current Research.

The survey by Dowsland and Dowsland [9] is one of the more complete reviews of the application of operational research techniques to the solution of two and three-dimensional packing problems such as pallet packing and container loading. As well as modelling and solving problems, the authors review a number of algorithmic and heuristic approaches. The emphasis of this work is practical solutions to real issues. Sweeney and Paternoster [8] review of the stock cutting and packing problem contains over 400 categorised application oriented references including books, dissertations and working papers. The authors have also grouped the publications according to the three main solution methodologies, these are summarised below:

- **Sequential assignment heuristics**, packing of patterns based on a set of assignment rules. The majority of heuristic approaches consist of determining what order and orientation the pieces should be packed in.
- **Single-pattern generating** procedures such as dynamic programming based algorithms, which try and reuse a single *optimal* packing configuration. For example in the two-dimensional rectangular packing problem, the solution is built up by considering partial solutions within smaller containing rectangles [9].
- **Multi-pattern generating procedures** such as linear programming based approaches, which consider the interaction between patterns. This approach requires the solutions to be rounded and are, therefore, also heuristic in nature [15]. The packing task can also be formulated as a binary integer-programming problem in which a single variable represents each possible shape position. A major concern with this approach is the production of a physical design from the values of the variables in the integer programming solution [9].

Dychoff [23] develops a consistent and systematic approach for integrating various kinds of cutting and packing tasks to try and unify the various concepts found in

---

[3] Also referred to as the *flyaway kit* problem by some authors [15].

the operational research literature. By doing so the author attempts to find the appropriate methods for each relevant problem type and conversely to identify problem types that can be solved by a certain method. A practical review of two and three-dimensional packing issues and solution methods can be found in Dowsland's [7] paper. The majority of the applications outlined in this review are based on two-dimensional packing techniques. Many of the three-dimensional problems are tackled by applying two-dimensional techniques on a layer by layer basis. Most published work in the area of three-dimensional packing is limited due to its complexity, and the applications that are discussed tend to be concerned with the loading of shipping containers. The paper also summarises some of the practical requirements in pallet loading, these include the stability of the loading stack, the load bearing ability of the items in the stack, ease of stacking and the air circulation requirements of certain products in a stack.

Dowsland [7] reviews some of the heuristic approaches used for packing a given set of identical rectangular items into a containing rectangle. A summary of the techniques used in the packing of non-identical rectangles is also included. This extensive review covers the key areas in automated packing, such as optimality versus efficiency and the measurement of a packing systems performance. The basic conclusion of the author is that although some very high packing densities have been reported in the literature, as yet there is no generic heuristic approach that can be applied to the two-dimensional packing task. Solutions reported tend to be very application specific.

Z.3.1 Packing of Regular Shapes.

The main emphasis of the early research into packing issues tended to concentrate on the well-constrained problem of packing regular shapes. This task usually consists of packing two-dimensional regular shapes into a well-defined scene, such as a rectangle [6,15,16,24,25,26,27]. The main industrial applications are in the area of pallet packing [24,25] and container loading [28]. Other applications include efficient VLSI design and automated warehousing [29,30]. Hall, Shell and Slutzky's [29] work combines automated packing techniques developed in the field of operational research, with systems engineering and artificial intelligence approaches to packing. It outlines the issues associated with the arrival of packages at the packing station and relates this to the single server queuing problem, which is commonly discussed in the operational research literature. The authors also discuss a number of systems issues, such as the importance of the product information. A practical example would be the packing of foodstuffs and toxic products. In this case the packing strategy has not only to consider the efficiency of the packing procedure, it must also consider the product type. The foodstuffs and the toxic products should be packed in different boxes. These boxes should be well separated on the pallet to prevent contamination of the food. The authors also highlight the importance of how the pallet data is represented and how to determine the correct placement location for the robot. Other related areas of research discussed include, bin-picking, automated storage and retrieval, automated kitting of parts for assembly, automated warehousing, and line balancing.

Bischoff [31] discusses the methodologies of the pallet and container packing problem. The main emphasis of this paper is a discussion of the techniques used in the interactive tuning of packing algorithms. The author points out that the pallet packing stability criterion is application dependent. If the pallet load is wrapped or strapped down, then this issue becomes less important. This is a significant point as there is often a conflict between stability constraint and need to minimise waste space on the pallet. The concept of 'cargo fragility' in container loading, and its relationship to the stability requirements, is also discussed.

Z.3.2 Packing of Irregular Shapes.

More recently researchers have begun to concentrate on the issues involved in the packing of irregular shapes. Batchelor [13] outlines a technique for the packing of irregular shapes based on the use of the minimum area-bounding rectangle. In this approach each shape is enclosed by its minimum fit bounding rectangle, and these rectangles are packed using the techniques developed for the packing of regular shapes. Qu and Sanders [32] discuss a heuristic nesting algorithm for irregular parts and the factors affecting trim loss. The application discussed is the cutting of a bill-of-materials from rectangular stock sheets. The authors take a systems approach to the problem and produce some good results. These are discussed in the context of performance measurements, which they have developed. While the authors review the published work in this area, they make the important point that although a number of techniques have been developed to enable the flexible packing of irregular shapes, very few of these have been published due to commercial confidentiality.

Qu and Sanders [32] describe irregular shapes in terms of a set of non-overlapping rectangles. The authors state that each of the parts in their study can be represented by no more than five non-overlapping orthogonal rectangles. The system places each part in an orientation such that (a) its length > height and (b) the largest complimentary (void) area is in the upper-right corner. The parts are then sorted by non-increasing part height. The shapes are packed into a rectangular scene in a raster fashion, building up layers of intermeshed packed shapes. The major disadvantage with this approach are (a) the use of rectangles to approximate the shape to be packed and (b) the assumption that good packing patterns will be orthogonal. Dori and Ben-Bassat [14] and Chazelle [33] were the first to investigate the nesting of shapes within a polygon rather than a rectangle. The authors discuss the optimal packing of two-dimensional polygons with a view to minimising waste. The algorithm is only applicable to the nesting of congruent convex figures. The problem involves cutting a number of similar but irregular pieces from a steel board, this is referred to as the *template-layout problem*. The authors decompose the task into two sub-problems. The first consists of the optimal (minimal waste) circumscription of the original irregular shape by the most appropriate convex polygon. The remaining problem consists of circumscribing the convex polygon by another polygon that can pave the plane, that is, cover the plane by replications of the same figure without gaps or overlap. This is referred to as the *paver polygon*. Limitations of this approach include the fact that it is only applicable to congruent convex figures and the assumption that the packing plane is infinite, hence waste in the margin is not considered. Another limitation of this approach is that it can only be applied to convex components with straight sides.

Koroupi and Loftus [34] address the issues raised by Dori and Ben-Bassat [14], by enclosing the component within a polygon so that the area added is minimal. The identical components, whether regular or irregular, are then nested using paving techniques. Martin and Stephenson's [35] paper deals with the packing of two and three-dimensional objects into a well defined scene. In this paper the authors tackle the task of automated nesting from a computer-aided design perspective. That is, given an arbitrary polygon and a rectangular box, will the polygon fit in the box, and if so how should the polygon be translated and rotated to implement this fitting. Prasad and Somasundaram [19] outline a heuristic based computer aided system that will allow the nesting of irregular-shaped sheet-metal blanks. This paper also contains a comprehensive list of the practical constraints one must consider in developing a packing system for sheet metal stamping operations. Constraints such as bridge width, blank separation, grain orientation, and the minimisation of scrap. They also highlight the need to align the pressure centre of the blank to be cut out with the axis of the press ram to reduce wear in the guideways of the press. Design requirements, such as maximising the strength of the part when subsequent bending is involved, are also considered.

Chow [36] discusses the optimal packing of templates of uniform orientation under limited conditions. This paper is useful as it discusses the edge effect issues in packing, which in general tend to be neglected. The author also outlines some of the concerns associated with manual packing. Kothari and Klinkhachorn [37] present a two-dimensional packing strategy capable of achieving dense packing of convex polygon shapes. The techniques described have been applied to the stock cutting in the hardwood manufacturing industry. This consists of efficiently cutting wooden pieces from a hardwood board so that the pieces are free of defects and aligned in the direction of the grain. This last constraint is needed for strength and aesthetic reasons. Albano and Sapuppo [38] outline a procedure, which will produce an optimal arrangement of irregular pieces. Manual and semi-automatic approaches to this nesting task are also discussed. The techniques described show how the optimal allocation of a set of irregular pieces can be transformed into the problem of finding an optimal path through a space of problem states from the initial state to the goal state. The search approach developed makes certain assumptions about the task; (a) the pieces are irregular polygons without holes and (b) that the scene is rectangular. The main application discussed is that of cloth layout and leather cutting.

Vincent [39] discusses the application of morphological techniques to the *tailor suit* or space allocation problem. This addresses the problem of translating two shape pieces, *A* and *B*, such that both are included in a larger shape piece *X* without overlapping. Although this can be shown mathematically for two pieces [39], there is no general solution to this problem involving simple morphological techniques only. While the author does not make the link between this technique and the automated packing and nesting in an industrial system, the use of such a powerful technique to manipulate shapes does seem to point the way forward.

Z.4 System Implementation

The packing scheme consists of two major components. The first is referred to as the *geometric packer*, and is based upon the principles of mathematical morphology. This component takes an arbitrary shape in a given orientation and puts the shape into place, *in that orientation*. A key element in the success of this approach is that it removes the limitations imposed by having to recognize and describe the object under analysis in order to pack it, thus increasing the systems flexibility. The second component is referred to as the *heuristic packer*, and is concerned with the ordering and alignment of shapes prior to the application of the *geometric packer*. This component also deals with other general considerations, such as the conflicts in problem constraints and the measurement of packing performance. In addition, it deals with practical considerations, such as the effects of the robot gripper on the packing strategy, packing in the presence of defective regions, anisotropy ("grain" in the material being handled) and pattern matching considerations.

By using heuristics in the packing strategy, it is hoped to produce an efficient, but not necessarily optimal solution. However, the main problem with such an approach is that there is a tendency to generate a set of overly complex rules, incorporating a variety of paradoxes and logical conflicts. It is necessary, therefore, to keep all the logic decisions as simple as possible. Another key aspect of applying heuristics to any complex problem is knowing when the solution is 'good enough' so that the process can be terminated and a result produced [40]. To this end, a mechanism for the measurement of the packing systems performance must also be included in the overall system design.

One of the key features in such a system is that it should work towards an efficient solution, accepting that we cannot guarantee reaching an optimal solution. Therefore a mechanism for quantifying the packing systems performance will be necessary. This will enable a quantitative comparison of packing procedures. Burdea and Wolfson [11] suggest that the integration of such a heuristic approach with a packing verification procedure should ensure convergence to an efficient solution. The general packing strategy outlined in this chapter is illustrated in Fig. Z.1 (a more detailed discussion of the system implementation can be found in [2,3]). Together the geometric and heuristic elements form a flexible strategy that allows the packing of arbitrary, two-dimensional shapes in a previously undefined scene. The aim of the design was to produce a flexible system capable of dealing with the majority of packing/cutting problems, and as such, the system was not designed around a specific application. As Burdea and Wolfson [11] point out, no single strategy, however efficient, will succeed in dealing with all shapes equally well. Therefore, when faced with a specific application, the system can be tuned to that task.

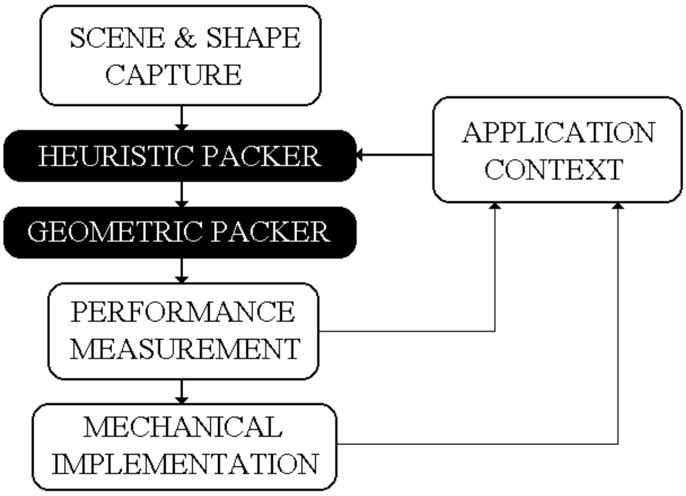


Figure Z.1. General packing strategy.

Z.4.1 Geometric Packer: Implementation.

The concept of enclosing an arbitrary shape within its bounding rectangle, convex hull or polygon approximation (known as paver polygons) is common to many of the irregular packing techniques discussed previously. Since these techniques involve the packing of the paver polygons rather than the original shape, only approximate packing solutions can be generated due to the loss in original shape information. Other strategies, such as contour matching, describe the shapes in terms of their critical points or chain codes. Although such approaches tend to be precise, they are also computationally expensive, especially for complex shapes. For this reason these techniques are rarely implemented on complex shapes without some degree of shape approximation. It would be advantageous to avoid using such estimates of the arbitrary shape. Therefore an approach that deals *directly* with shapes would be of great benefit in the development of a flexible packing strategy. Such an approach can be found in the set-theoretic concepts of mathematical morphology [41,42] which is concerned with the extraction or imposition of shape structure. One of the key features of the application of morphological operations to automated packing is that the shape to be packed and its scene do not have to be formally described to enable their manipulation by the packing system.

The function of the geometric packer is to take any arbitrary shape in a given orientation and to put that shape into place in the scene, efficiently in that orientation. Providing the shape(s) to be packed, and the scene to be examined, can be captured and stored as binary images then morphological techniques can be applied to these images. These techniques allow the packing of a structuring element into a given image scene. In the case of the automated packing system, the shape to be packed will be represented by a morphological structuring element, while the scene will be represented by an image set on which this structuring element will act.

In the sample geometric packing problem illustrated in Fig. Z.2(a), the image scene is a rectangular bounded region (in which a *star* shape has already been packed) and is denoted by the image set A. The *star* shape to be packed, is applied to the image set A and is denoted by the structuring element B. The image scene A is eroded by the structuring element B, to produce the erosion residue image C. Every white pixel in this residue represents a valid packing location. The erosion residue image is then scanned, in a raster fashion, for the location of the first (white) pixel. This location is denoted by *(fitx, fity)*. Experimentation has shown that further erosion of the residue image C by a standard 3x3 square structuring, prior to searching for the first packing location, enables control of the spacing between the packed shapes. That is, the number of pixel stripping operations, on the erosion residue, is related to the spacing between the packed shapes. This relationship can also be shown mathematically [43]. The translation of the shape to be packed, B, to the location *(fitx, fity)* effectively places B at the co-ordinate of the first possible packing location of B in the scene A. This image is denoted by $B_{(fitx,\ fity).}$ The resultant image is subtracted from the original image set A to produce a new value for the image set A, therefore effectively packing B into the scene (Fig. Z.2(b)). This procedure can then be reapplied to the image set A until an attempt to pack all the input shapes have been made (Fig. Z.2(c)).

Z.4.2. Heuristic Packer: Implementation.

As outlined earlier, the *heuristic packer*, is concerned with the ordering and alignment of shapes prior to the application of the *geometric packer*. The *heuristic packer* operates on two classes of shapes: blobs (shapes with a high degree of curvature and/or significant concavities) and simple polygons. Details of these procedures can be found in [2]. It is necessary to consider both these general shape classes separately, since no single scheme exists for all shapes, and while the *geometric packer* is independent of the shape class and application context, the *heuristic packer* is not. The heuristic component also deals with other general considerations, such as the conflict in problem constraints and the measurement of packing performance. In addition, it deals with a number of practical issues, such as the effects of the robot gripper on the packing strategy [44], packing in the presence of defective regions [2,45], anisotropy and pattern matching considerations.

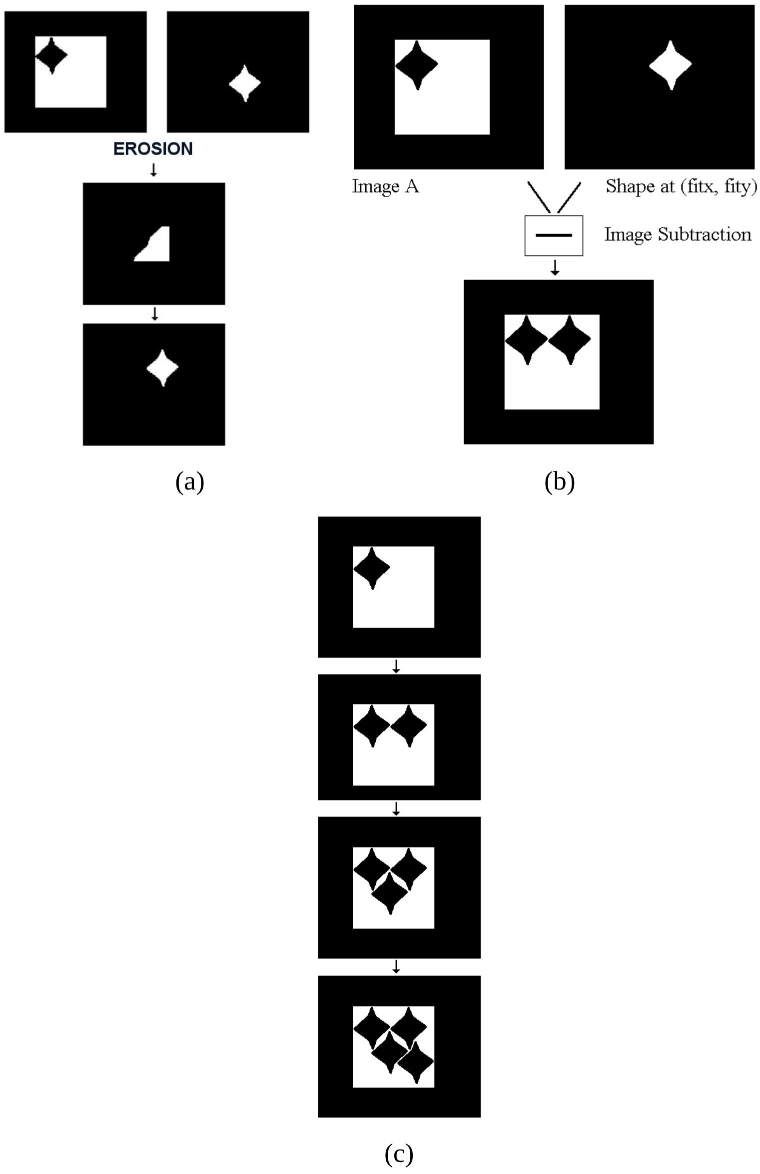


Figure Z.2. Geometric packing. This illustrates the steps involved in packing a star shape in a rectangular scene. (a) Image A is eroded by star shape to produce an erosion residue. (b) The star shape is then relocated to *(fitx, fity)*. (c) This process is repeated until a terminating condition is reached.

Blob Packing

This section outlines some of the heuristics that have been devised to deal with two-dimensional binary images of random shape and size, prior to the application of the geometric packer. The approach outlined was designed specifically for off-line packing but the techniques developed could equally well be applied to an on-line packing application.

All the shapes to be packed are presented simultaneously to the vision system. The shapes are then ranked according to their bay sizes; the shape with the largest bay is the first to be applied to the geometric packer. Once the shape ordering has been decided, it is necessary to orientate each shape so that an efficient local packing strategy can be implemented. Four orientation rules are used to align the shape to be packed in the scene.

The order in which the shapes are placed by the packer is determined by the *sort_by_bay* predicate defined below. If the area of the largest bay is significant compared to the area of the current shape, then the shape is sorted by its largest bay size (largest first). Otherwise the shapes are sorted by their size (largest first). The *bay_rot* predicate rotates a shape such that the largest bay is aligned with the scene's angle of least moment of inertia. This predicate also ensures that the biggest bay is facing into the scene (that is facing to the right and upwards). The operation of this predicate is summarised below:

- If *object_Y_coordinate* > *bay_Y_coordinate* then rotate shape by 180°
- If *object_Y_coordinate* = *bay_Y_coordinate* and *object_X_coordinate* *bay_X_coordinate* then rotate shape by 180°
- If *object_Y_coordinate* = *bay_Y_coordinate* and *object_X_coordinate* *bay_X_coordinate* then no action required as in correct orientation
- If *object_Y_coordinate* < *bay_Y_coordinate* then no action required as in correct orientation

Fig. Z.3(a) shows the result of packing hand tools into a rectangular tray. These shapes were initially presented directly to the geometric packer, without the aid of the heuristic packer. This has the effect of packing each tool at whatever orientation it was in when it was presented to the vision system. Fig. Z.3(b) shows the resultant packing configuration when the heuristic packer precedes the geometric packer; each shape is aligned and ordered, before it is applied to the geometric packer. Fig. Z.3(c) shows the packing of the tools into a "random" blob region. The full packing strategy was used again here, as in Figs. Z.3(b). Fig. Z.3(d) and (e) illustrates the packing of some general items, such as scissors, keys and pens, into a rectangular tray and an irregular scene using this approach.

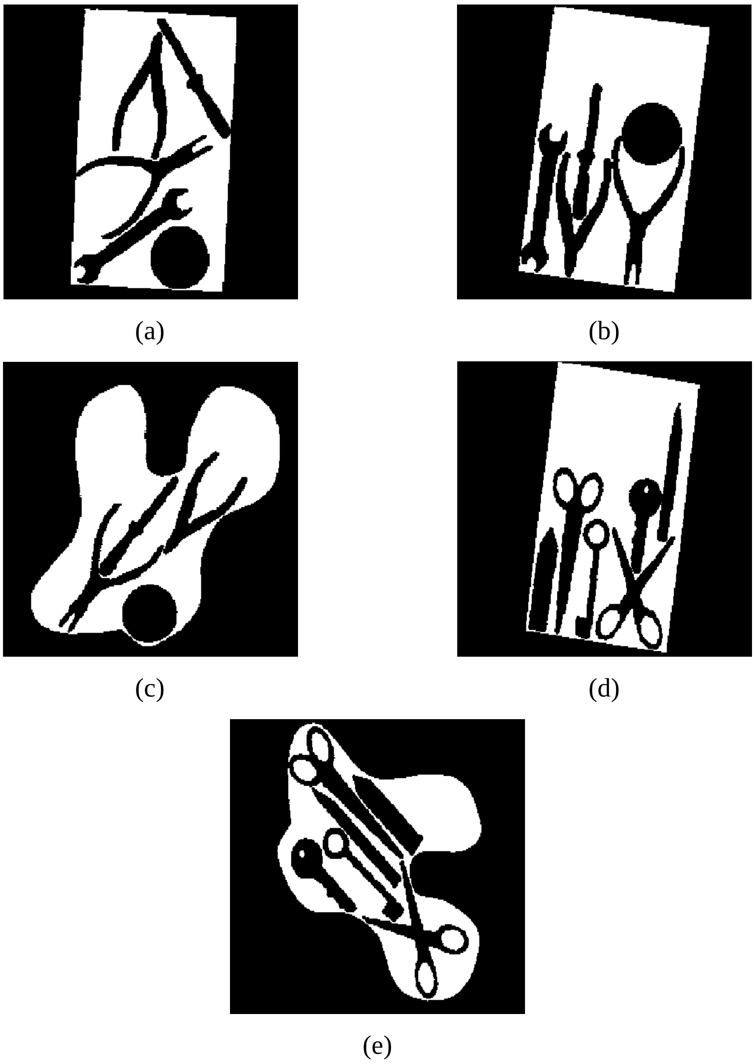
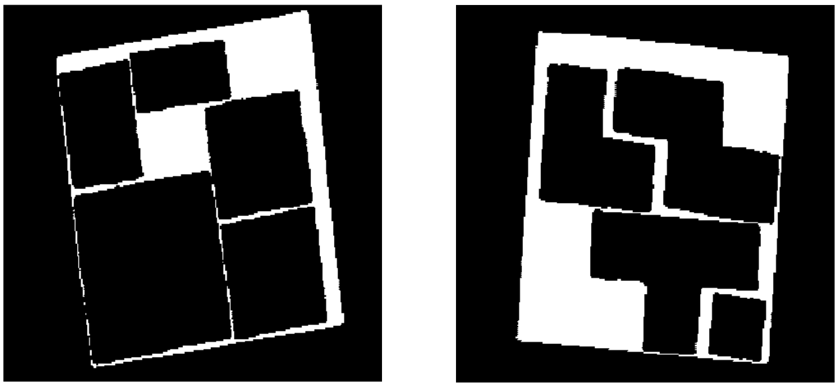


Figure Z.3 Automated packing implementation. (a) Tools packed in their current orientation. (b) Tools reorientated for better efficiency. (c) Tools packed in an irregular scene. Packing general items into: (d) a rectangular tray and (e) an irregular scene.

Polygon Packing

The previous approach is not efficient, when packing shapes which do not contain bays of significant area. Hence, a different packing procedure is used to pack simple polygons, which do not possess large bays. As before, this procedure was designed to work within an off-line packing system but could also be applied to on-line packing applications. Unlike the previous approach, however, this second procedure has the ability to determine the local packing efficiency for each shape and will reorientate it, if necessary, to ensure a more efficient configuration. (This local efficiency check could also be applied to the blob packing strategy.) In the second sample application, we chose to pack non-uniform box shapes (squares and rectangles) into a square scene (Fig. Z.4(a)). Once all the shapes have been presented to the packing system, they are ordered according to size, with the largest shape being packed first. The shapes must then be orientated, prior to the application of the geometric packer.



(a) (b)

Figure Z.4. Automated packing of simple polygons. (a) Non-uniform boxes in a square tray. (b) Block polygons.

In the initial versions of this packing procedure, each shape was aligned in such a way that its axis of least moment of inertia was matched to that of the scene under investigation. However, this method proved unreliable for packing squares, because quantisation effects produce a digital image, with a jagged edge. (An image resolution of 256x256 pixels was used.) Furthermore, a square has no well-defined axis of minimum second moment. This can cause errors in the calculation of the moment of inertia. The problem was overcome by aligning the longest straight edge of the shape to be packed with the longest straight edge of the scene. The edge angles for the shape and scene were found by applying an edge detection operator, followed by the Hough transform. The latter was used, because it is tolerant of local variations in edge straightness. Once the peaks in the Hough transform image were enhanced and separated from the background, the largest peak was found. This peak corresponds, of course, to the longest straight edge within the image under investigation, whether it is the shape or the scene. Since the position of the peak in Hough space defines the radial and the angular position of the longest straight edge, aligning the shape and the scene is straightforward.

Once a polygonal shape has been packed, a local packing efficiency check is carried out. This ensures that the number of unpacked regions within the scene is kept to a minimum. The shape to be packed is rotated through a number of predefined angular positions. After each rotation, the number of unpacked regions in the scene is checked. If a single unpacked region is found, then a local optimum has been reached. In this case, the local packing efficiency routine is terminated and the next shape is examined. Otherwise, the local packing efficiency check is continued, ensuring that, when a shape is packed, a minimum number of unpacked regions exist. This reduces the chance of producing large voids in the packed scene, and improves its overall efficiency of packing.

The packing order is determined by the sizes of the shapes to be packed (largest first). The rotation of the shapes by the packer is based on the angle of the largest face (longest straight side of the polygon) of the unpacked region. The predicate *shape_face_angles* finds the largest face angle and stores it in the face angle database. This database also contains a selection of rotational variations for the current shape. The face angles are sorted such that the angle of the largest face appears at the top of the database. The other entries are modified (by a fixed angle rotation factor) versions of this value. The *blob count* refers to the number of "free space blobs", that is the number of blocks of free space available to the packer. The polygon packer operates according to the following rules:

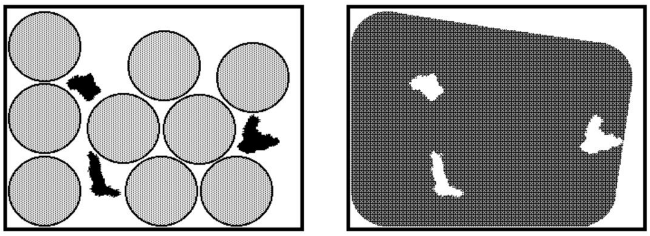
- If *blob count* is 1 then the best fit has occurred, so exit and view the next shape.
- If *blob count* is 0 then read the new angle from face angles database and retry.
- If *blob count* < *local optimum* then update blob count and update the local optimal storage buffer before trying the next angle in the face angles database.
- If *blob count* ≥ *local optimum* then try the next angle in database.

Z.5 Performance Measures

To ensure that we have confidence in the global efficiency of any packing strategy, there must be some way of measuring its performance. Traditionally, packing performance has been measured by a single number, called the *packing density* [46]. This is the ratio of the total area of all the packed shapes to that of the total area of the scene. This is referred to as the *worst-case analysis* packing measure.

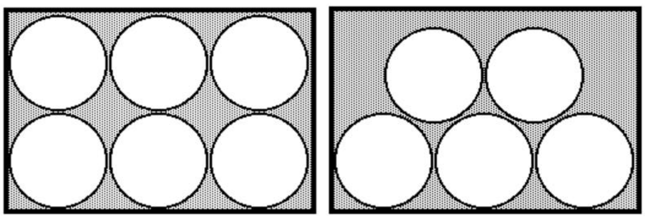
A number of other performance measures have been developed in the field of operational research, particularly for comparing different heuristics for packing rectangular bins by odd-sized boxes [7]. These performance metrics fall into two main categories: *Probabilistic* and *statistical* analysis [47]. While these performance measurements can be quite useful in well-constrained packing problems, they are of little use in dealing with the packing of arbitrary shapes. Since it is unlikely that real data will fall neatly into a uniform, or any other easily analysable distribution. The performance measures used in our strategy are based on the traditional *worst-case analysis*. After a packing procedure has been applied to a given scene, the result is assessed by a number of performance parameters [2].

- *Packing density* is the ratio of the total area of all the shapes packed, to the area of their (collective) convex hull after packing (minus the area of the scene defects). This measure has a maximum value of 1. (Fig. Z.5)
- The *performance index* is a modified version of the packing density in which a weighting factor is applied. This is referred to as the *count ratio* and is defined as the ratio of the total number of shapes packed, to the number of shapes initially presented to the scene. The *performance index* is equal to the product of the *packing density* and the *count ratio*. The *performance index* also has a maximum value of 1. This measure accounts for any shapes that remain unpacked when the procedure terminates. (Fig. Z.6)



(a) (b)

Figure Z.5. *Packing density* calculation. (a) The *approximated optimal* packing area is calculated by summing the area of the individual packed shapes (indicated by the shaded disks). The black blob regions indicate scene defects. (b) The *actual* packing area is denoted by the area of the convex hull of all the shapes packed in the scene (minus the area of any scene defects). This area is indicated by the shaded region in (b).



(a) (b)

Figure Z.6. Quantitative comparison of packing configurations using, *packing density*, *performance index* and *count ratio* values. (a) In this packing configuration all six shapes presented to the rectangular scene were packed, giving a *count ratio* of 1. Therefore the *performance index* equals the *packing density* (calculated as 0.77). (b) In the second configuration only five of the six shapes presented were packed, resulting in a *count ratio* of 0.83. Although the *packing density* for this configuration is better, at 0.82, the *performance index* is only 0.68, due to the fact that not all the shapes were packed.

Z.6 System Issues

In general, the design of the packing system can be greatly simplified the more application constraints that can be incorporated into the heuristic packer. The design of packing is made easier by the fact that many natural materials have a pronounced grain. For example, in certain applications only two orientations of a given shoe component may be permissible, a fact which can greatly enhance the speed of the packing procedure. Again, the heuristic packer can easily take this type of application constraint into account. Alternatively, some practical considerations can increase the complexity of the packing procedure.

Z.6.1. Packing Scenes with Defective Regions.

Any practical automated packing system for use in such industries as leather or timber processing must be able to pack objects into a scene, which may contain defective regions. The heuristic packer can readily accommodate defective regions; by simply defining the initial scene to contain a number of holes. Fig. Z.7(a) illustrates the effect of packing tools into a rectangular tray which contains four small blob-like defects. By comparing this to the packing configuration shown in Fig. Z.3(b), it is clear that the packing is not as compact when defects are taken into account. Fig. Z.7(b) shows the packing of jacket template pieces on to a piece of fabric, prior to cutting. The small blob-like regions indicate the defective areas in the fabric. These defective regions are not to be included in the jacket pieces to be cut. These results illustrate the flexibility of the packing strategies adopted, when applied to the automated cutting of natural materials.

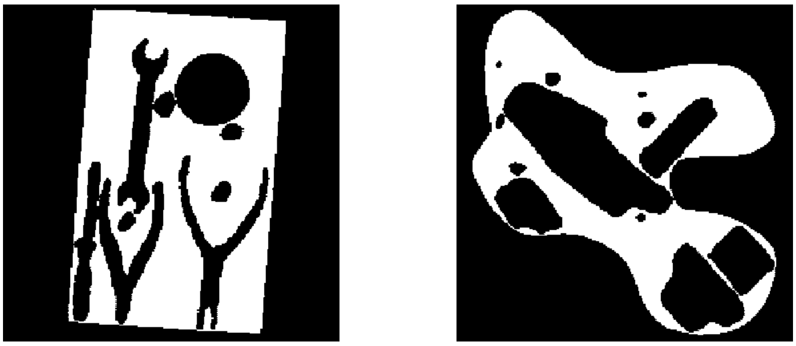
Z.7 Packing of Templates on Leather Hides.

The purpose of the application outlined in this in this section, is to automatically arrange, and place, shape templates on to an arbitrary shaped non-homogenous leather hide in an efficient manner (so as to minimize the leather waste). The importance of good packing procedures in the leather industry is obvious, since the raw material is both expensive and non-recyclable[4]. Presently, as much as 40% of the hide is wasted

---

[4] In general, any waste material produced in the cutting of leather hides is sold to external companies who deal in small leather goods.

[48], and the European Union funded ALCUT project aimed to reduce this waste by up to 8%. This represents a significant saving for any firm dealing with a large hide turnover, for example the fashion industry in which a new range of footwear is introduced to the market about twice a year.



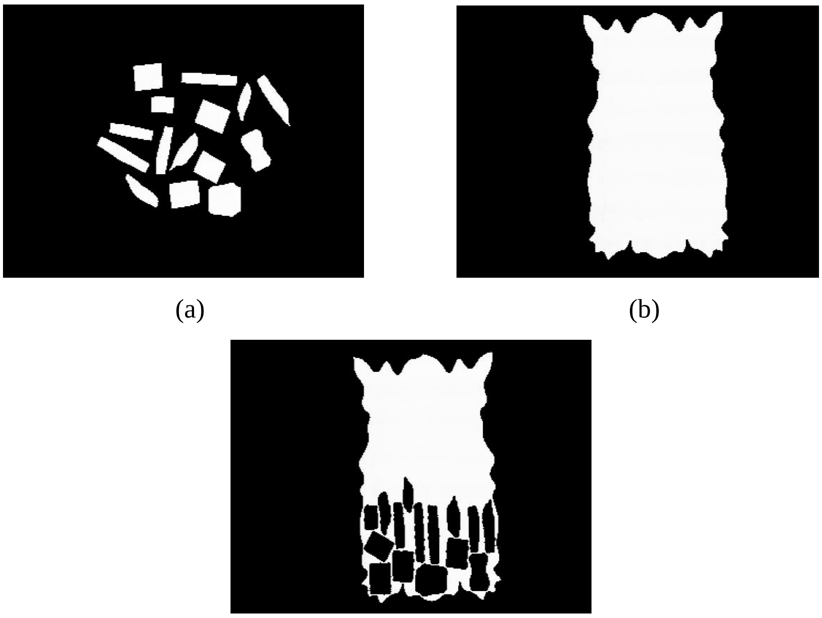
(a) (b)

Figure Z.7. Packing items into defective regions. (a) Packing tools into a defective tray. (b) Cutting jacket template pieces from a fabric segment, which contains defective regions

In the current generation of leather cutting systems the hides pass underneath a bank of line scan cameras, which generates a two-dimensional image of the hide. The region of the hide to be cut is then determined by the placing of shape templates on the scanned hide (generally, the shape templates are entered into the CAD system via digitising tablets or external databases). The placing of the shape templates is done interactively by trained operators at CAD workstations. Once the position of the shape templates have been finalized, the corresponding regions are automatically cut from the hide [20]. The size and shape of these templates are application dependent. The specific application addressed in this discussion concerns the cutting of shape pieces for use in the upholstery of high quality leather car seats.

The CAD operators have only a short time period in which they can place the shape templates on the hide. This is the time between when the hide is imaged by the vision system and when the hide has progressed underneath the cutting station (in leather upholstery the operators will have to deal with 35-40 different shapes over a global surface of 55 to 60 ft^2 [49]). To this end, semi-automated interactive CAD systems have been developed to aid the operator in maximising (a) the speed at which the shape templates are placed on the hide, and (b) the number of shape templates to be cut from a given hide. The interactive functions allow operations such as bumping, sliding, automatic repetition of shapes, quality matching, grouping and area filling. The main packing strategy used by these operators involves the packing of the larger template pieces on the outer edges of the hide, and progressively moving in towards the centre with the remaining shapes.

The approach outlined in this chapter has been used to automate the template layout process (i.e. the automatic layout of the hide template shapes in an *efficient* manner). Initial investigations have concentrated on the application of the unmodified packing strategies outlined previously. The application of the *polygon* packing procedure to the automatic placement of the template pieces on a leather hide, produces an efficient packing configuration (Fig. Z.8). This highlights the flexibility of the approach taken. (A medium resolution CCD array camera was used in the prototype system to capture the hide and shape template images. This resulted in quantisation errors on rotation of the shape template pieces. A full-scale system requires high-resolution line scan cameras to build up the hide image.)



(a) (b)

(c)

Figure Z.8, Automatic placement of car seat template pieces. (a) Car seat template pieces. (b) Leather hide image. (c) The resultant packed image using the *polygon* packing procedure. The *polygon* packing strategy is implemented since there are no significant bay regions in the shapes to be packed.

Z.7.1 Packing of Templates on Defective Hides.

Any practical automated packing system, for use in such industries as leather, textile or timber processing, must be able to pack "objects" into a scene that may contain defective regions. In a natural hide there are a number of regions that cannot be used (The ideas outlined can also be applied to the cutting of synthetic leather, these are produced as more regular shapes). These consist mainly of the spine and corner regions, although defects can occur anywhere within the hide. Currently, the hide is marked using chalk or removable inks, to indicate the stress directions and to aid in defect and quality recognition during the hide scanning process. The automated packer can readily accommodate defects like these; by initially defining the scene to contain a number of holes. Fig. Z.9 illustrates the packing of leather templates onto such a hide. The black blob-like regions, illustrated in Fig. Z.9(a), indicate the defective areas of the hide. These defective regions are not to be included in the leather pieces to be cut.

While the packing systems ability to automatically image the shape templates is of little benefit in this application (since the shapes are generally stored in databases and do not change frequently) the automated packing procedures do enable a more adaptive packing strategy to be implemented. The procedures ability to deal with random shapes and previously undefined defective regions, without the need for software modifications or human interaction, is a significant benefit of this approach.

(a) (b)

Figure Z.9. Automatic placement of car seat template pieces illustrated in Figure 8(a) on to a defective leather hide. (a) The black blob like regions indicate the hide defects. (b) The resultant defective hide after the shape templates have been packed.

Z.7.2 Additional Points on Packing in the Leather Industry.

The design of packing systems for the automated cutting of leather hides is made easier by the fact that leather, like fabric, wood, marble and many other natural materials has a pronounced grain or stress direction. This means that quite often only two orientations of a given leather component are permissible, a fact which can greatly enhance the speed of the packing procedure. Again, the *heuristic packer* can easily take this type of application constraint into account. Packing leather component templates onto a hide is not quite as simple as suggested, because the leather is not uniform in its thickness and suppleness. When making shoes, for example, the components which will make up the soft leather uppers are cut from the stomach region of the hide, while the tougher, more rigid sole is take from the back. Adding heuristic rules to assist packing under these constraints should not difficult, although this has not yet been attempted.

Further complications arise from the fact that natural leather hides contain a number of quality levels (or grades). Each region of a shape, depending on its importance and visibility, must satisfy a *quality matching criteria*. The hide is subdivided into several areas of constant average quality. The shapes are also given a well-defined quality, therefore each single shape, or part of a shape, can only be positioned on a portion of the hide with the *same or higher quality level* [20]. In the cutting of shape templates for the manufacture of high quality leather furniture, there may be up to 40 grades of leather, whereas in the application discussed above (high quality car seats) there are only 5 grades. One objective of such a layout system is to keep high quality parts of the hide for those components of the object which are the most visible and to try and utilize lower quality regions for non-visible parts. For example, in the cutting the leather component of a car seat armrest, some of the leather will not be exposed to the driver, and as such it can be of a lower leather grade. This also influences the speed of the cutting operation, since lower quality parts are cut less precisely and at a higher speed. Therefore, not only do the leather pieces have to be packed to minimise waste, but the template grades must be positioned to suit the available grades of leather on a given hide. This has not yet been implemented on the system.

Z.8 Conclusion

The work outlined in this chapter was motivated by the need to produce a new generation of flexible packing/cutting systems. The approach adopted is capable of implementing efficient packing strategies, with no prior knowledge of the shapes to be packed or the scenes into which the shapes were to be placed. Automated packing systems have a wide range of possible industrial applications, including flexible assembly and automated cutting systems.

The strengths of the adopted approach become more evident when the systems issues of a specific application are considered. The packing system outlined, has the ability to deal with a range of such issues. These include the ability to pack shapes into defective regions. This is not a trivial task for a human operator. Other issues that must be considered include the ability of the automated packing procedure to control the spacing between packed items in a consistent manner. This is a task that manual operators would find difficult, especially for irregular shapes about which they had no prior information.

Z.9 References

[1] H.L. Dreyfus and S.E. Dreyfus, *Mind over Machine*, The Free Press, 1986.

[2] P.F. Whelan and B.G. Batchelor, "Flexible packing of arbitrary two-dimensional shapes", *Optical Engineering*, 32(12), pp 3278-3287, 1993.

[3] P.F. Whelan and B.G. Batchelor, "Automated Packing Systems - A Systems Engineering Approach", *IEEE Trans. on Systems, Man and Cybernetics - Part A: Systems and Humans*, 26(05), pp 533-544, 1996.

[4] I. Stewart, "How to succeed in stacking", *New Scientist*, pp. 29-32, July 1991.
[5] I. Stewart, "Has the sphere packing problem been solved ?", *New Scientist*, May 1992.
[6] E.L. Hinrichsen, J. Feder and T. Jossang, "Random packing of disks in two dimensions", *Physical Review A*, Vol. 41, No. 8, pp. 4199-4209, 1990.
[7] W.B. Dowsland, "Two and three dimensional packing problems and solution methods", *New Zealand Operational Research,* Vol. 13, No. 1, pp. 1-18, 1985.
[8] P.E. Sweeney and E.R. Paternoster, "Cutting and packing problems: A categorized, application-orientated research bibliography", *Journal of the Operational Research* Society, Vol. 43, No. 7, pp. 691-706, 1992.
[9] K.A. Dowsland and W.B. Dowsland, "Packing problems", *European Journal of Operational Research,* Vol. 56, pp. 2-14, 1992.
[10] H. Wolfson, E. Schonberg, A. Kalvin and Y. Lamdan, "Solving jigsaw puzzles by computer", *Annals of Operations Research,* Vol. 12, pp. 51-64, 1988.
[11] G.C. Burdea and H.J. Wolfson, "Solving jigsaw puzzles by a robot", *Proc. IEEE Trans. On Robotics and Automation,* Vol. 5, No. 6, pp. 752-764, 1989.
[12] S.R. Oh, J.H. Lee, K.J. Kim, and Z. Bien, "An intelligent robot system with jigsaw-puzzle matching capability", *Proc. 15'th Int. Symp. On Industrial Robots (Tokyo),* pp. 103-112, 1985.
[13] B.G. Batchelor, *Intelligent Image Processing in Prolog,* Springer-Verlag, London, 1991.
[14] D. Dori and M. Ben-Bassat, "Efficient nesting of congruent convex figures", *Communications of the ACM,* Vol. 27, No. 3, pp. 228-235, 1984.
[15] A.R. Brown, *Optimum Packing and Depletion: The Computer in Space- and Resource-Usage Problems,* American Elsevier Publishing, 1971.
[16] H.L. Ong, M.J. Magazine and T.S. Wee, "Probabilistic analysis of bin packing Heuristics", *Operations Research,* Vol. 32, No. 5, pp. 983-998, 1984.
[17] M. Hofri, "Bin packing: An analysis of the Next-Fit algorithm", *Technical Report #242,* Israel Institute of Technology, 1982.
[18] A.K. Chandra, D.S. Hirschberg and C.K. Wong, "Bin packing with geometric constraints in computer network design", *Operations Research* 26(5), 760-772, 1978.
[19] Y.K.D.V. Prasad and S. Somasundaram, "CASNS - A heuristic algorithm for the nesting of irregular-shaped sheet-metal blanks", *Computer-Aided Engineering Journal* 8(2), 69-73, 1991.
[20] S. Dulio, "Application of automation technology to leather cutting", *Proceedings of the 23'rd International Symposium on Automotive Technology and Automation*, 83-96, 1990.
[21] D. Chetverikov and A. Lerch, "Prototype machine vision system for segmentation of hide images", *International Journal of Imaging Systems and Technology* 4(1) 46-50, 1992.
[22] R. Cuninghame-Green, "Geometry, shoemaking and the Milk Tray problem", *New Scientist* (12 Aug), 50-53, 1989.
[23] H. Dyckhoff, "A typology of cutting and packing problems", *European Journal of Operational Research*, Vol. 44, pp. 145-159, 1990.
[24] H. Carpenter and W.B. Dowsland, "Practical considerations of the pallet-loading problem", *J. Opl. Res. Soc.*, Vol. 36, No. 6, pp. 489-497, 1985.
[25] C.S Chen, S. Sarin and B. Ram, "The pallet packing problem for non-uniform box sizes", *International Journal of Production Research*, Vol. 29, No. 10, pp. 1963-1968, 1991.
[26] F.R.K. Chuang, M.R. Garey and D.S. Johnson, "On packing two-dimensional bins", *SIAM J. Alg. Disc. Meth.*, Vol. 3, No. 1, pp. 66-76, 1982.

[27] B.S. Baker, E.G. Coffman and R.L. Rivest, "Orthogonal packing in two dimensions", *SIAM J. Comput.*, Vol. 9, No. 4, pp. 846-855, 1980.
[28] E.E. Bischoff and M.D. Marriott, "A comparative evaluation of heuristics for container loading", *European Journal of Operational Research* Vol. 44, pp. 267-276, 1990.
[29] E. Hall, R. Shell and G. Slutzky, "Intelligent packing and material handling", *Proc. SPIE Intelligent Robots and Computer Vision IX: Algorithms and Techniques,* Vol. 1381, pp. 162-170, 1990.
[30] R.C. Wilson, "A packaging problem", *Management Science*, Vol. 12, No. 4, pp. B135-B145, 1965.
[31] E.E. Bischoff, "Interactive approaches to packing problems", *10th ICPR*, pp. 55-61, 1989.
[32] W. Qu and J.L Sanders, "A nesting algorithm for irregular parts and factors affecting trim losses", *Intl. Journal of Prod. Res.*, Vol. 25, No. 3, pp. 381-397, 1987.
[33] B. Chazelle, "The polygon containment problem", *Advances in Computing Research*, Vol. 1, pp. 1-33, 1983.
[34] F. Koroupi and M. Loftus, "Accommodating diverse shapes within hexagonal pavers", *Int. J. Prod. Res.*, Vol. 29, No. 8, pp. 1507-1519, 1991.
[35] R.R. Martin and P.C. Stephenson, "Putting objects into boxes", *Computer-Aided Design,* Vol. 20, No. 9, pp. 506-514, 1988.
[36] W.W. Chow, "The packing of a template on a flat surface", *Trans. of the ASME - Journal of Mechanical Design,* Vol. 102, pp. 490-496, 1980.
[37] R. Kothari and P. Klinkhachorn, "Packing of convex polygons in a rectangularly bounded, non-homogeneous space", *IEEE Proc. of 21st Southeastern Symp. on System Theory*, pp. 200-203, 1989.
[38] A. Albano and G. Sapuppo, "Optimal allocation of two-dimensional irregular shapes using heuristic search methods", *IEEE Transactions on Systems, Man, and Cybernetics,* Vol. 10, No. 5, pp. 242-248, 1980.
[39] L. Vincent, "Morphological transformations of binary images with arbitrary structuring elements", *Signal Processing*, Vol. 22, pp. 3-23, 1991.
[40] E.A. Silver, R.V.V. Vidal and D. de Werra, "A tutorial on heuristic methods", *European Journal of Operational Research* **5**, 153-162, 1980.
[41] E.R. Dougherty, *An Introduction to Morphological Image Processing*, Tutorial Text TT9, SPIE Press, 1992.
[42] P.F. Whelan and D. Molloy, *Machine Vision Algorithms in Java: Techniques and Implementation*, Springer (London), 298 Pages. ISBN 1-85233-218-2, 2000.
[43] R.M. Haralick and L.G. Shapiro, "Mathematical Morphology", Chapter 5 of *Computer and Robot Vision: Volume 1*, Addison-Wesley Publishing Co. Ltd, 1992.
[44] P.F. Whelan and B.G. Batchelor, "Development of a vision system for the flexible packing of random shapes", in *Machine Vision Applications, Architectures, and Systems Integration*, Proc. SPIE, Vol. 1823, pp 223-232, 1992.
[45] P.F. Whelan and B.G. Batchelor, "Automated packing systems: Review of industrial implementations", in *Machine Vision Applications, Architectures, and Systems Integration II*, Proc. SPIE, Vol. 2064, pp 358-369, 1993.
[46] R. Fowler, M. Paterson and S. Tanimoto, "Optimal packing and covering in the plane are NP complete", *Inf. Proc. Letters*, Vol. 12, No. 3, pp. 133-137, 1981.
[47] M.R. Garey and D.S. Johnson, *Computers and Intractability - A Guide to the Theory of NP-Completeness*, W.H. Freeman and Co, 1979.
[48] EUREKA (1989), *Robotics and Production Automation*, European Community.
[49] S. Dulio (1992), Private communication.