Template Matching in Support of Generalisation of Rural Buildings

Desmond Rainsford and William Mackaness

Geography Department, The University of Edinburgh, Drummond St, Edinburgh EH8 9XP, 0131 650 8163, wam@geo.ed.ac.uk

Abstract

A range of methodologies has been proposed to derive generalised forms of buildings at coarser scales. In this paper, we explore the use of simple pattern matching algorithms in order to select from a set of templates, a building outline that best characterises a more detailed form. This template matching process is used to simplify the form of rural buildings (farmsteads) in Danish mapping, and is currently a manual task within the National Mapping Agency of Denmark (KMS). This research has explored the feasibility of automating this approach, and reports on its implementation, and provides an evaluation. The challenge in pattern matching is to minimise the misidentification of patterns (type I error) and failures to find any match (type II error) in the assignment of templates. The initial set of results was very encouraging.
Keywords: building generalisation, pattern recognition, automated cartography

1 Building Generalisation

Levels of generalisation are controlled by the map scale, theme and the importance attached to map objects. While manual generalisation was entirely appropriate for the creation of paper products at fixed scales in a pre-digital age, it is an impediment to the rapid production of electronically derived maps at arbitrary scales. It is usually still necessary to maintain separate databases of different levels of detail in order to produce maps of the same area at different scales or for different themes. Such a constraint is hugely inefficient in terms of data maintenance and storage. With good reason, efforts at automated generalisation have focussed almost entirely on emulating the skills of human cartographers. Curiously, little attempt has been made to use automated techniques to change and perhaps improve upon the cartographic conventions that have evolved with paper maps over the centuries. It could be argued that the subject of this project is a case where a better solution could be not to try to mimic a traditional map making technique, but to adopt a more flexible approach offered by automated simplification methods.

Litchner (1979) was one of the earliest workers to successfully demonstrate an automated approach specific to building generalisation. Using seven "elementary" procedures (simplification, enlargement, displacement, combination, selection or elimination, classification or exaggeration) he was able to produce a satisfactory generalisation of city buildings. However, his approach did not take into account context. Rule-based approaches to building generalisation have since become more prevalent. Edwardes et al. (1998) demonstrates the analysis of cartometrics measures (squareness, compactness etc.) to create a set of rules that are used to select procedures to effect generalisation. Glover and Mackaness (1999), on the other hand, devised a rule set based on display scale, map type and object class). This paper explores an alternative approach based on template matching – inspired by manual methods used by the Danish Mapping Agency KMS for the generalisation of rural buildings. A broad range of techniques exist for both pattern matching (Veltkamp 2001a; Veltkamp 2001b; Loncarid 1998), and automatic feature extraction from remotely sensed images (for example the work by Stassopolou et al. (2000) on building detection). This paper builds on ideas of pattern matching which show favourable comparison more conventional approaches to map generalisation.

2 Shape Recognition

Farm buildings in Denmark are displayed on small scale maps, by representing them using a series of simple alphabetic templates with similar shapes. The use of templates in effect creates caricatures of the farm building outlines. This approach has been adopted by the KMS to provide visual consistency in their small scale maps such that meaning of these features can be understood from their shapes. At such scales (1:50,000), it can be argued that the implied meaning of a small feature is more important to the user than its representational accuracy. Thus, it is common to see the generalisation of buildings such as churches represented by a uniform symbol. Templates, on the other hand, preserve more of the character of the physical object. Fig. 1 shows a small sample from a KMS product showing the original building in grey and the generalised form (a fitted template) in outline.

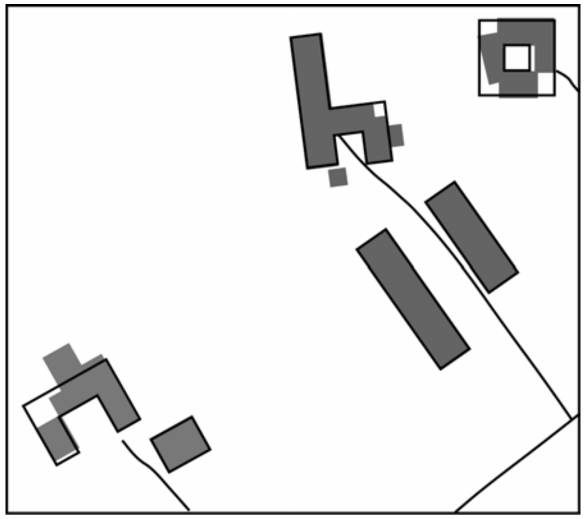


Fig. 1. Manual template matching

The aim of this project was to automate a process which is presently carried out manually by the Danish Mapping Agency. In order to perform this task, nine polygon templates have been created (identified by the alphabetical characters they resemble), based on a specification document provided by KMS. A template matching process requires the recognition of polygon shapes as well as the application of simplification procedures. Shape recognition has not been a central theme in map generalisation, although the use of "descriptive trees" and "minimum spanning trees", for this purpose, are described by Weibel and Dutton (1999). Most of the literature on this subject exists outside of the geographic realm. A review of shape characterising procedures by Ehler *et al*. (1996) cites single values measures (e.g. circularity and form), fourier measures (for sinuous lines), image content queries (IBM development) and the use of binary shape matrices. The first of these requires the judicious use of several measures but is relatively easy to implement. The fourier method is not well suited for polygonal shapes. Image content queries we will discuss later. Binary shape matrices are used to fit shapes in the raster domain by the application of a roving spinning filter. In principal, this approach could be applied to the current task, but would require the rasterization of the building shapes and would be complicated to apply.

Perhaps the richest vein of pattern recognition expertise is related to image analysis. One technique, skeletonisation, provides a synthetic and thin representation of objects that are useful for the description of shape (Attali and Montanvert, 1997). While skeletonisation is a powerful technique for extracting topology from shapes, its use is beyond the scope of this exercise. Also of interest is the use of Hausdorff-based image comparison techniques using the generalised Hausdorff measure. This technique has been shown to be effective for recognising partly obscured targets in photographic images and is obviously well suited for surveillance applications. Doubtless, this technique could be used for shape selection, but it would be an overly complex for the scope of problem addressed here.

3 Methodology

Because we are dealing with single building polygons only, we do not have to address the problems of aggregation and displacement and we can concentrate on the steps required for simplification. The national mapping agency of Denmark (KMS) has a set of defined templates that are used in the manual process of characterising farmsteads. The dimensions of each template are altered to fit those of the farm buildings that it represents. In practice, this means the template shapes are most often stretched or flattened to achieve satisfactory results. For the purpose of this research, a subset of templates was modelled (illustrated in Fig. 2).

Fig. 2. The templates: I F P G E L U O T

The template fitting scheme uses a two step process of simplification followed by a template selection procedure. This is figuratively shown in Fig. 3. The latter operation progressively narrows the choice of templates, based on the number of holes in the object, the number of vertices and the sequence of internal angles, until a unique selection is arrived at.

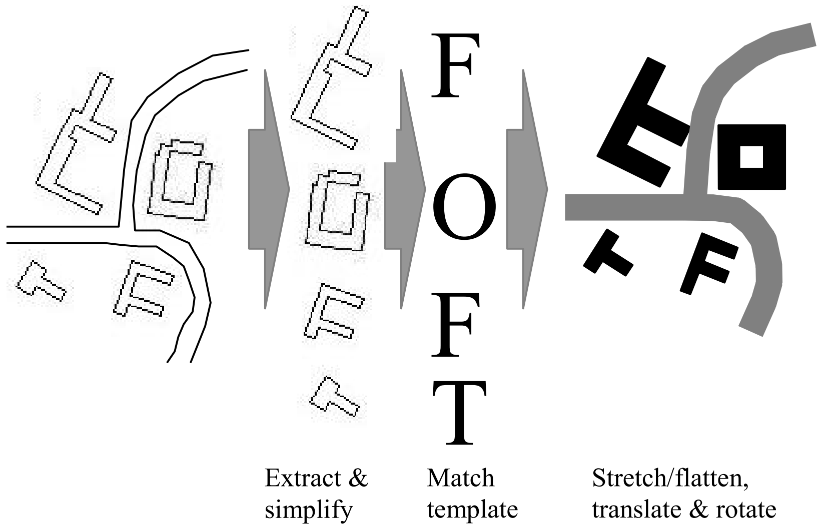


Fig. 3. The complete process

3.1 Simplification

The simplification part of the algorithm begins by checking that at least one dimension is 25 metres or greater. If not, the building is ignored on the basis that, at 1:50,000 scale, it would be less than 0.5 mm and so not worth plotting. The shapes are then squared and buffered followed by reduction to close any open loops. Minor appendages (“nubs”) are removed and this is followed by fitting a Minimum Bounding Rectangle (MBR) which is used to determine the parameter required for the second round of simplification.

A study was made of template geometries in order to establish rules that would specify when to delete “limbs” on a building shape. Conceptually, we wish to eliminate limbs that are less than half the length of those on the templates. Because the templates are not fixed in size we needed to examine the ratio of limb lengths to template dimensions. It was found that the most reliable measure was the ratio of limb length to template width (MBR width). It appeared that the limb cut-off threshold should lie in the range 0.18 to 0.36 of the MBR width.

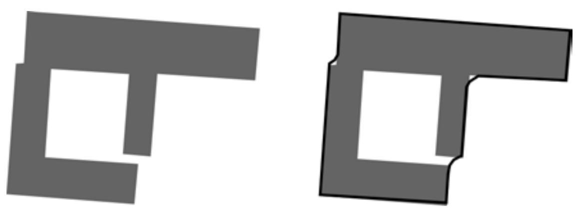
Initially the simplification algorithm used a fixed length of 7.5 metres, which translates to 0.15 mm at 1:50,000 scale. It was subsequently found that a range of values could be used, the figure becoming simpler in form as the simplification tolerance was increased. In the evaluation phase, we noted that, in several instances, a larger factor would sometimes result in a fit being made where none had been obtained before but that the match was erroneous. (i.e. the larger the tolerance, the less likely that the match would be correct). This simplification tolerance provided a simple index of how likely that the match was correct. Starting by simplifying with a small factor (0.15 * MBR width) an attempt is made to see if a template could be fitted. If the shape is unmatched it is further simplified by applying a factor of 0.20 * MBR width and attempting again to match it. A final attempt is made with the simplification factor increased to 0.25 * MBR width. If no match is found at this attempt, “NO MATCH” is returned and no additional simplification is carried out lest the shape is oversimplified and a template is wrongly selected.

3.2 Selection

Some building shapes have central courtyards or 'holes' in them. It is a simple process to count the number of boundaries or 'rings' that a feature has. Knowing the number of rings, we select either template groups I, L, U, T, F, E (1 ring) or O, P (2 rings).

3.3 Closing rings

Some features look like closed courtyards, but were frequently not closed structures (such as Fig. 4a). In order for these building shapes to be matched by one of the 2 ring templates it was necessary to close them. An algorithm that counted the number of rings of an object, coupled with a 'shrink wrap' convex hull algorithm, enabled us to assess how close a building was to being a closed courtyard and thus enable us to categorise a feature like the one in Fig. 4a as a '2 ring' feature (one ring describing the perimeter and an additional ring shown as the dark outline generated using a shrink wrap convex hull function.



(a) (b)

Fig. 4. A farm building with a central courtyard and an open loop **(a)** is processed using a 'shrink wrap' convex hull function. Comparing the two footprints enables the courtyard to be identified.

Using the number of vertices, the program selects within one of these groups and returns either a single candidate or more. The next stage is to examine the angle sequence that characterises the shape in order to differentiate within the '1' or '2' ring group.

3.4 Angle Sequencing

Internal angles of the simplified and squared polygons were measured and recorded in order to characterise their shapes. Within tolerance, the angle should either be 90^0 or 270^0. We deem 90^0 to be a "left" turn and store this information as an "L" in a string. In the same way 270^0 is deemed to be a "right" turn and an "R" is stored. Thus, we end up with a word string describing the angle sequence of the shape (e.g. **RLRRLLRRR**). This string is then passed to our **match_sequence** function.

3.5 Sequence Matching

The purpose of the **match_sequence** operation is to match the measured angle sequence with those of a selected template. As we can make no assumptions as to where in the shape we started recording the angles, we need to cycle the angle sequence and attempt a match with the template sequences until either a match has been found or the sequence is completely cycled. The process is illustrated in Fig. 5.

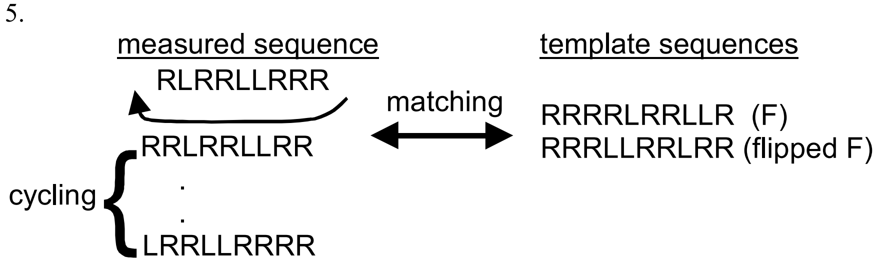


Fig. 5. Schematic representation of the operation of the match_sequence function

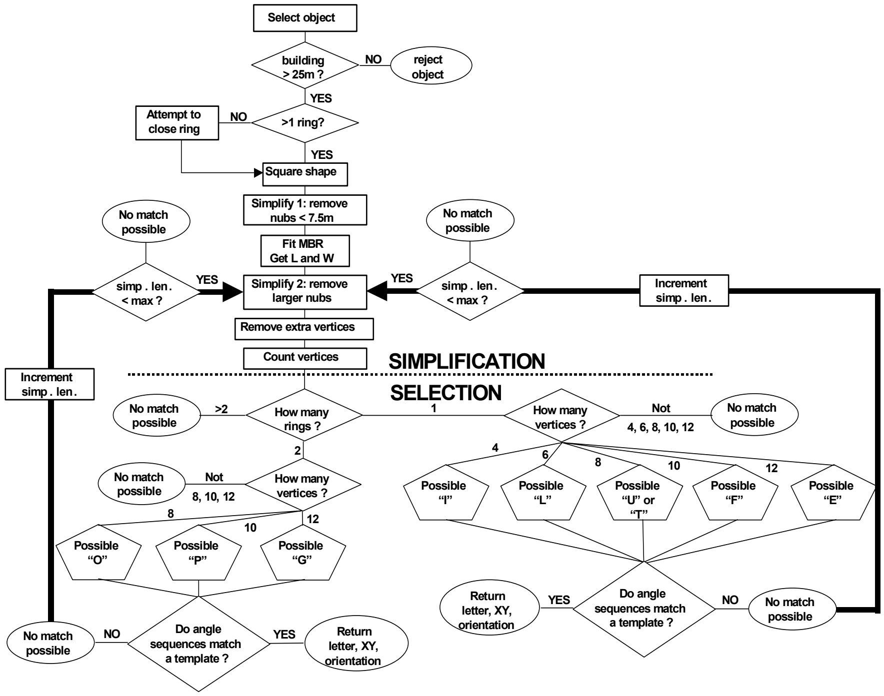


Fig. 6. Workflow of the template matching algorithm

As we cannot assume the facing direction of the template, we include angle sequences for the templates facing forward and backwards. We are then able to return the facing direction of the shape as well as the template ID. This information is, of course, only useful for non-symmetric templates (e.g. "F" and "G"). This final test reduces the candidates to a single template (if there is more than one) and eliminates spurious shapes which happen to have the same number of vertices as the candidate template(s). The entire process is summarised in the flow diagram illustrated in Fig. 6. Fig. 7 shows the transformation stages and sequencing process for an example building.

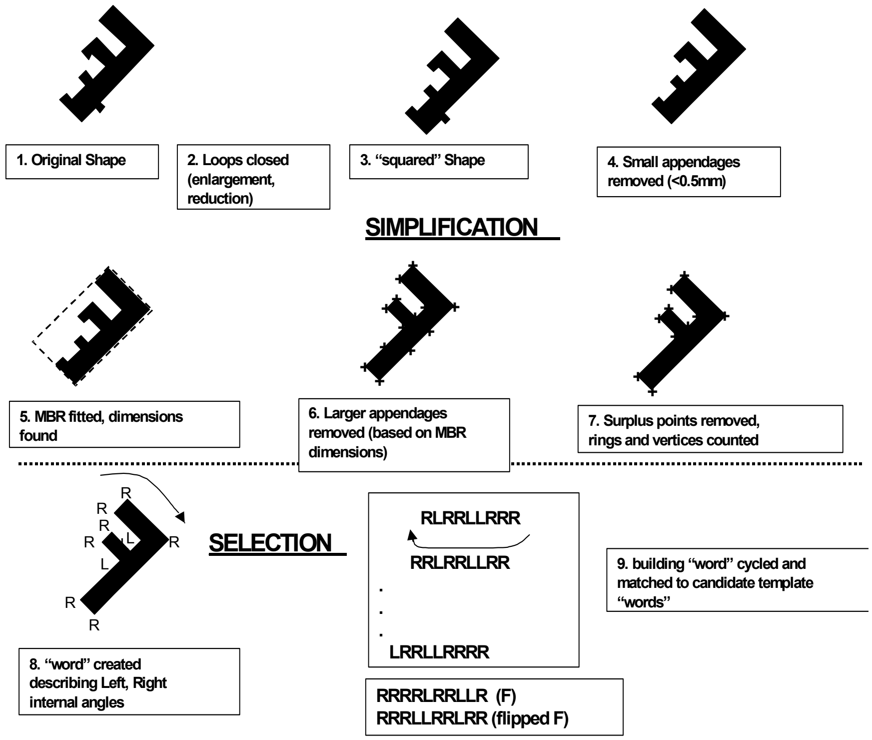


Fig. 7. The various processing stages in identifying an 'F'

Fig. 8 shows diagnostics from the algorithm and reveals how the amount of simplification is iteratively increased in the search for the best template match. In this example, the first iteration attempts to match the shape unsuccessfully to an "F". Additional simplification in the second iteration results in the successful match of the shape to a "U". In this case, because a "U" is symmetrical, its facing direction is meaningless. The results are displayed in the last two lines of the printout. Fig. 9 shows the transformation that took place in the simplification process as part of the diagnostics detailed in Fig. 8.

```
number of outer vertices = 10
number of holes = 0
Attempting to close ring...
First round of building simplification..
building was NOT simplified
Fitting bounding box...
Second round of building simplification..
Iteration 1:
calculated simplification length = 8.650238:
final number of outer vertices (geom_id)= 10
Getting angle sequence..
i: 1  angle sequence: L
i: 2  angle sequence: LR
i: 3  angle sequence: LRL
i: 4  angle sequence: LRLL
i: 5  angle sequence: LRLLL
i: 6  angle sequence: LRLLLL
i: 7  angle sequence: LRLLLLR
i: 8  angle sequence: LRLLLLRR
i: 9  angle sequence: LRLLLLRRL
i: 10  angle sequence: LRLLLLRRLL
Begining template selection..
First pass selection of candidate templates is: 'I', 'L', 'U', 'T', 'F', or 'E'
Second selection of candidate templates is: 'F'
Matching angle sequence to 'F'
Iteration 2:
calculated simplification length = 11.533651:
building was simplified
final number of outer vertices (geom_id)= 8
Getting angle sequence..
i: 1  angle sequence: L
i: 2  angle sequence: LL
i: 3  angle sequence: LLR
i: 4  angle sequence: LLRR
i: 5  angle sequence: LLRRL
i: 6  angle sequence: LLRRLL
i: 7  angle sequence: LLRRLLL
i: 8  angle sequence: LLRRLLLL
Beginning template selection..
First pass selection of candidate templates is: 'I', 'L', 'U', 'T', 'F', or 'E'
Second selection of candidate templates is: 'U' or 'T'
Matching angle sequence to 'U'
match found!!!
The selected template letter is: U, Facing: forward
Centroid (X, Y) = 570390.606220, 6246590.148621   Orientation (radians) = 1.75035
```

Fig. 8. Diagnostics from the algorithm in the search for a best fit

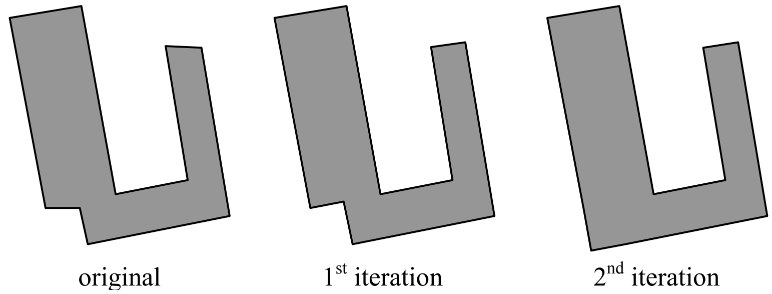


Fig. 9. Simplification of a building outline leading to the selection of a "U" template. The sequence illustrates the processes detailed in Fig. 8

4 Results

In the evaluation phase, we assessed each building shape and subjectively assigned it a template letter. Because this was a subjective process, we recognised that some of the assessments were questionable and that in some cases it was appropriate to suggest more than one answer because of ambiguity (Fig. 10).

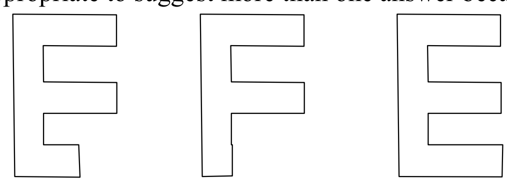


Fig. 10. Which is it: an E or an F?

Fig. 11 shows the results of the matching. In each box is an outline of the farmstead building, and two sets of letters. The lower letters are what we subjectively deem correct solutions. The upper letter is the template proposed by the algorithm. The program was 'right' if it produced a result that was the same as our subjective assignment (black building), 'wrong' if an incorrect template was identified (white building) and 'unclassified' if no template was returned if a reasonable match existed (shaded building). 'No' means no template was found (upper letter) or no template was expected to be found (lower letter). We define the 'wrong' and 'unclassified' results as type I and type II errors respectively. A summary of the testing is displayed in Fig. 12. By opting to iterate through three levels of simplification we were able to improve the success rate of template matching about 3% to 78% with 15% wrong answers (the remainder being unclassified). We also kept a record of the facing directions returned for the asymmetric templates. The results were completely accurate for the "F" template, random for the "P" template and the absence of suitable shapes did not allow the "G" template to be tested. The inability of the program to detect the proper facing direction of "P" shapes was due to the fact that the angle sequencing took into account only the outer ring. The angles of the outer ring of a "P" are the same as that of an "L" shape, which is symmetrical and cannot therefore be faced.

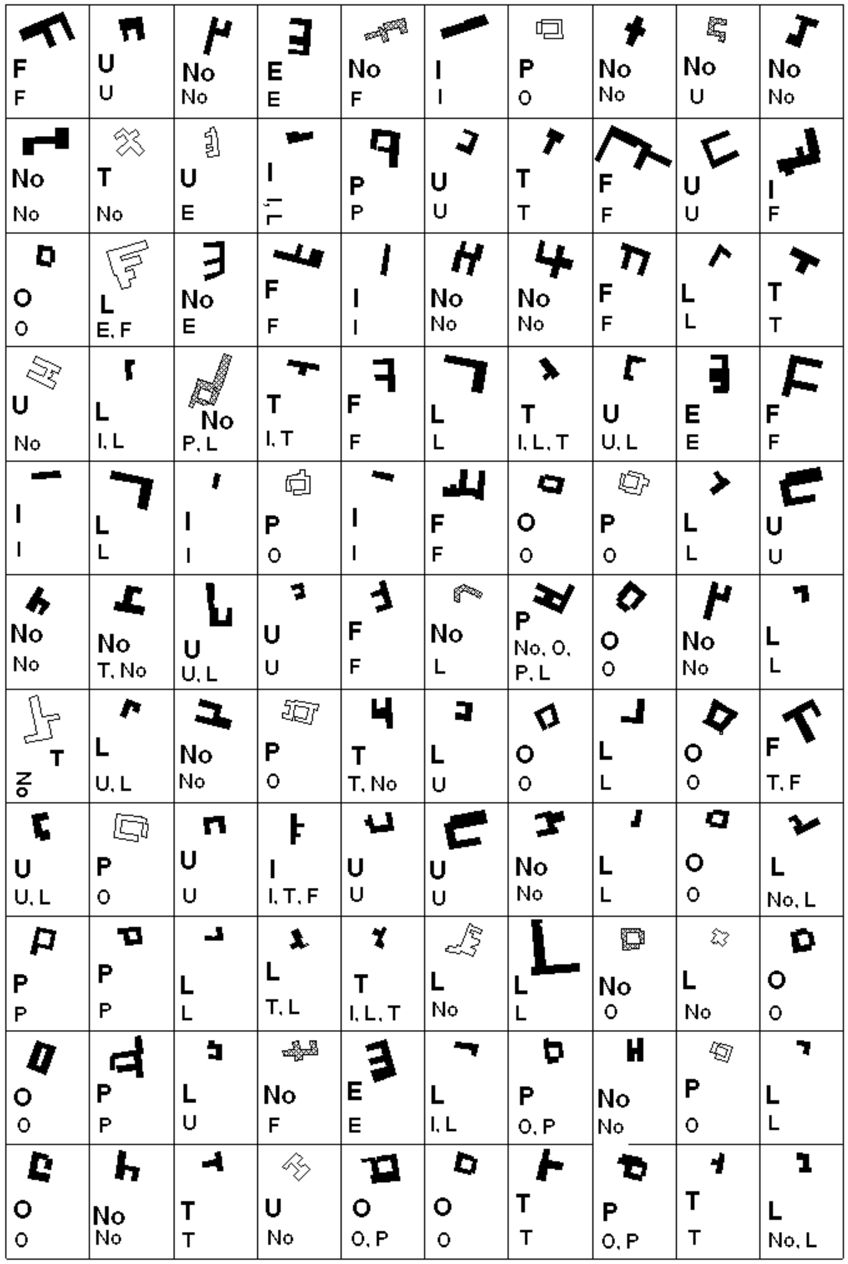


Fig. 11. Testing of a subset of farm buildings

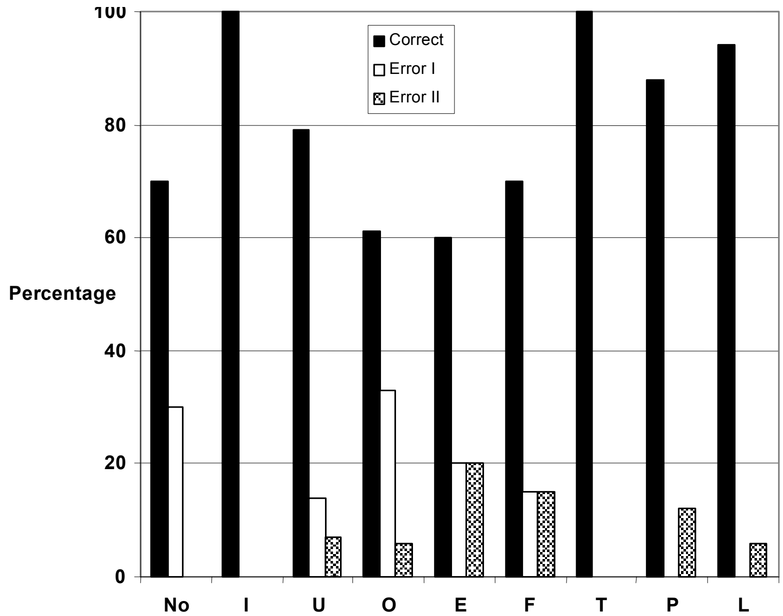


Fig. 12. Degrees of success for each of the eight templates modelled

5 Evaluation

When we examine the results in detail and in particular the failures, we noted some deficiencies in the algorithm and saw how some improvements might be made. These are examined below.

5.1 "O" Shapes that are Identified as "P's"

Several of the failures were as a result of irregular "O" shapes that were insufficiently simplified and were identified as "P's" instead. We attempted to rectify this by increasing the simplification parameter, but this resulted in other shapes being oversimplified and mis-identified as a result. Instead, a better alternative might have been to apply a higher level of simplification specifically to those shapes that were found to contain a courtyard.

5.2 Buildings with More than One Hole

The algorithm assumes that, if the building has one inner ring, no further rings need to be closed. This may not always be the case, as a farm could conceivably have two or more courtyards. Ideally, all shapes should be tested regardless of whether they already have a closed courtyard.

5.3 Lack of Templates

Owing to the fact that only eight templates were considered, the algorithm would often try in vain to find an appropriate template. With each iteration and associated higher level of generalisation, the risk of oversimplification increases. If more templates were introduced, matches could be made earlier in the process and the risk of misidentification through oversimplification reduced.

6 Conclusion

The template matching scheme, outlined in this document, has the advantage that it is fairly simple to implement and uses a small number of measures to identify the shapes. The scheme should be able to unambiguously detect confounding shapes and reject these. The solution is also scalable and additional templates can easily be incorporated by including their angle sequences, in the appropriate program branches, after counting vertices.

As far as limitations are concerned, the scheme is highly dependent on a satisfactory simplification being obtained before vertices, loops and angles are measured. Although rules have been devised for the simplification process, further testing will be required to see if these are adequate. While a squaring method is invoked, it is expected that the program will be less effective in matching shapes with angles markedly different from right angles.

Without too much additional effort the templates could be displayed in the place of the farm building. Our algorithm already returns all the information needed to accomplish this. Knowing the identity of the template, a copy could be made, flipped (if required), rotated and translated to the shape centroid. In addition, using the dimensions of the MBR, that we had obtained in the course of the simplification process, we could scale the template in X and Y so its size and aspect approximates that of the farm building. This step is required so that simple transformations in x and y can be used stretch and flatten accordingly. Further refinement and extension of the range of templates is required before the system can be commercialised. Reducing the occurrence of mis-matches (type I errors) to a very low level would be highly desirable even if it were to require modifications that reduced the number of successful matches as well. In such an automated environment, where user intervention is required, it is important to minimise the effort of the operator and to direct them to buildings for which a template has not been found. The algorithm provides a simple measure of template matching quality, through the incremental increase in the simplification tolerance. From the operator perspective, templates could be colour coded according to how quickly a solution was reached.

Acknowledgments

The researchers are grateful to Laser Scan for the use of their software, and to the National Mapping Agency of Denmark for the use of their data, and the template specification document on which this work was based.

References

Attali D, Montanvert A (1997) Computing & Simplifying 2D & 3D Continuous Skeletons. Computer Vision & Image Understanding 67:261-273

Edwardes A, Regnauld N, Mackaness W (1998) A rule based approach to algorithm selection in building generalisation. Internal AGENT report EDIN-001-10-05-1998

Ehler GB, Cowen DJ, Mackey HE Jr. (1996) Development of a shape fitting tool for site evaluation. In: Advances in GIS Research II: Proceedings 7th International Symposium on Spatial Data Handling. Delft, I:4A pp 1-12.

Glover E, Mackaness WA (1999) Dynamic generalisation from single detailed database to support web based interaction. 19th International Cartographic Conference, vol 2. pp 1175-1183

Litchner W (1979) Computer Assisted Processes of Cartographic Generalization in Topographic Maps. Geo-Processing 1:183-199

Loncaric S (1998) A survey of shape analysis techniques. Pattern Recognition 31(8):983-1001

Stassopolou A, Caelli T, Ramirez R (2000) Building Detection using Bayesian Networks. International Journal of Pattern Recognition and Artificial Intelligence 14(6):715-734

Veltkamp RC (2001a) Shape Matching: Similarity Measure and Algorithms. In: Proceedings Shape Modelling International. 7-11 May, 2001, Genova, Italy, IEEE Press, pp 188-197

Veltkamp RC, Hagedoorn M (2001b) State-of-the-art in shape matching. In: Lew M (ed) Principles of Visual Information Retrieval. Springer Verlag, pp 87-119

Weibel R, Dutton G (1999) Generalising spatial data and dealing with multiple representations. In: Longley P, Goodchild, MF, Maguire DJ, Rhind, DW (eds) Geographical Information Systems. John Wiley, New York, pp 125-156