Document Feature Recognition using a mesh of Associative Memories

S. E. M. O'Keefe[*], J. Austin[†]
Advanced Computer Architecture Group
Department of Computer Science
University of York
Heslington
York YO1 5DD

Abstract

This paper describes a new approach to the problem of identification of complex objects in document images. The novelty of the approach lies in its use of a distributed representation for objects which seeks to overcome some of the problems of data noise and incompleteness. Objects are modelled as a set of features and the relationship between neighbouring features. Feature recognition is performed by a mesh of associative memories. Each feature in the image stimulates recall from an associative memory of the other local features. Evidence is accumulated locally, and the most likely label is assigned to each feature. The associative memories then communicate to update the labels synchronously until a stable, locally consistent labelling of image features is obtained. Thus, incomplete image data may lead to recognition and recall of a complete object in a translation-invariant, robust manner. The consistency requirement effectively suppresses noise, significantly reducing the false positive rate. The architecture is designed for analysis of large binary images, for example fax images (typically 1143×1728 pixels). The mesh of associative memories is ideal for parallel implementation.

1 Introduction

Identification of complex objects in images is a problem which is at the heart of image analysis and computer vision [1, 2]. It is an essential ingredient of document image analysis, a field which has been studied extensively from the point of view of the identification of low level features [3, 4, 5, 6, 7, 8], and from the point of view of identification of structure within documents from the complex objects which have been identified in the image [9, 10, 11, 12]. However, the typical approach is to identify the areas which contain the text of a document, and segment these, apply some OCR technique and store the resulting text as ASCII characters, while the rest of the document is discarded or stored as images without further attention to the content. Thus little attempt is made to make use of the information which may

[*] sok@minster.york.ac.uk

[†] austin@minster.york.ac.uk

BMVC 1996 doi:10.5244/C.10.37

be present in the "non-text" areas of a document image, such as logos, trademarks, etc. The main difficulties in making use of the non-textual data in the image lies in the wide range of forms which it may take, and the computational requirement of identifying large components of the image (of the order of 100^2 pixels) by matching them against a database which may contain hundreds of models. This task is further complicated by noise in the image, and incompleteness of the objects to be identified. In previously reported work [13, 14, 15, 16], we have described a basic approach to the solution of these problems, involving the application of a technique similar to the Generalised Hough Transform (GHT).

The GHT [17, 18] is an analysis technique which relies on the accumulation of evidence for objects in an image via the association of object features with a set of parameters describing the object. The effectiveness of the GHT is limited by the size of the accumulators required (and therefore the amount of memory required) and problems with feature extraction and quantisation [19]. The main difficulties stem from inaccurate feature estimation and spurious features generated by noise in the image.

The GHT is an attractive algorithm for image analysis because it uses low level feature information in the detection of large scale, complex objects, bringing the information together in parameter space to provide an estimate of the class of object present. In the usual GHT, the feature information is transformed into a single point or block of points in parameter space, and it is the accumulation of these points which indicates the maximum likelihood for the object parameter values. This translation from feature space into parameter space removes any reliance on the connectedness of object edge elements, making the technique more robust that segmentation techniques which rely on edge-detection (for example [20]). However, when we have a large database of models against which we wish to match the data in the image, the GHT presents a computational problem. A separate parameter space for each class of objects would require a prohibitive amount of memory. The work described in [13, 14, 15, 16] presented a method for overcoming this problem, by using a compact coding for feature and object labels, and a mesh of associative memories for performing recall without a linear search of the object space. The basic architecture was described in [13], and the ability of the architecture to detect multiple instances of objects in an image, and to detect objects with added noise, was demonstrated in [14, 15, 16]. However, the overall performance in terms of the fraction of objects correctly recognised was adversely affected by quite low levels of noise, and objects of different sizes could not be reliably detected. The present paper focusses on the quantification of performance, and on the improvements in performance delivered by the modifications described here.

One of the problems encountered when applying an evidence accumulation scheme to the identification of an object with one of a large number of different classes, each of which has a different amount of "evidence" in the form of component features, is that the amount of evidence required to be confident of an assignment to a particular class is dependent on the class itself. This problem has been addressed in the work described here by the addition of the necessary information to the object model. Also, the generation of a single most probable value for object parameters discards much of the information present in the image in the form of the parameters of features making up the object. This has been addressed by implementing a modification to the GHT whereby subunits of the object are matched against the models, adjacent subunits communicating information about the probable feature and object classes. Feature and object labels are synchronously updated until a stable, locally consistent labelling is obtained.

We are particularly interested in the application of the recogniser to the analysis of document facsimile images. That is, we are interested in the segmentation of the fax into its components, the identification of each component, and the determination of the structural relationship between each of the components. Fax images have their own particular characteristics which need to be addressed, particularly their size and the characteristics of the noise generated by the scanning and transmission processes. Baird [7] has done work to characterise the noise which plagues scanned documents, and we have examined performance against a subset of the sources of noise appropriate to fax images.

The outline of this paper is as follows. In section 2, the architecture of the object recogniser is outlined. Section 2.1 briefly describes the GHT, and section 2.2 details the modifications used to overcome the problems inherent with the basic method. In section 3, the results of experiments are presented indicating their effect on object recognition. Finally, section 4 offers some conclusions.

2 Implementation

In this section, we give a brief description of the Generalised Hough Transform which forms the basis for the object recogniser. We then describe the enhancements to the basic system which allow for the recognition of multiple classes of objects simultaneously, and which improve the ability of the system to identify objects in the presence of noise. Details of the implementation of the basic recogniser are given in [13].

2.1 Generalised Hough Transform

The implementation of the recogniser is based on the GHT[17], which is briefly described here. The GHT provides, for each class of objects, a template in the form of parameters relating each feature to a reference point for the object. When a feature is detected in an image, a look-up is performed to find out which objects the feature may be a part of, and, for these objects, the position of the object centre relative to the feature. This information is then used to update counts of the number of features which have "voted" for an object centre at each point in the image. This is illustrated in figure 1. The system has been trained to recognise object classes **L1** and **L2**. A feature is detected at position **f**. This feature may be part of either **L1** or **L2**. From the look-up we recall that the centre for object **L1** should be at position **v1** relative to the feature, and object **L2** should be at position **v2**. We cast a vote for an instance of the object **L1** at **f+v1**, and a vote for **L2** at **f+v2**.

Peaks in the number of votes accumulated correspond to the position of objects. The advantage of the GHT over other template based methods is that, in order to recognise an object which is large relative to the image, we only need to

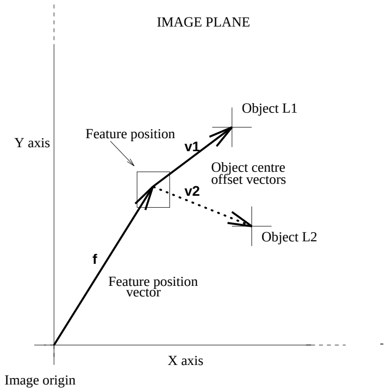

Figure 1: Principle of the GHT. The object is parameterised in terms of vectors between each feature and an object reference point. This can be visualised in image space as the feature casting a vote at the location of the object reference point.

recognise the individual features of the object (a much easier task than attempting to recognise the whole object) and accumulate the evidence from all the features. We are also able to recognise objects which are occluded, because the features which *are* visible will still vote together for the object. The recognition of the object is translation invariant.

2.2 Extensions to the basic system

The basic GHT as described above brings together all information from the features to a single point. When parts of the image are unclear because of noise or occlusion, the response of the GHT is reduced, and we have a lower confidence in the labelling of objects. However, when parts of the image are very clear and we can be very confident in their identification, we would like to use this information to make predictions about the parts of the object which are unclear, and to increase the overall confidence in the labelling of objects. We have modified the GHT algorithm to recall a distributed representation of the object, in which each feature is associated with its neighbouring features. Effectively, each group of features is recognised independently at first.

Instead of voting at a single point in parameter space representing the object (in our case the position and class of the object), each feature votes for the set of features which are close to it in parameter space. Here, this corresponds to features which are close to it in image space. The set of features which are voted for is defined by the size of the "local support neighbourhood" (LSN). Thus, a set of features which comprise an object and which are in the correct spatial relationship will all support each other. The votes accumulated for each feature are thresholded to determine the most likely label for each feature. The threshold appropriate for each feature depends of the context in which it occurs. That is, the structure of the objects of which the feature is a part will determine how many votes will be cast for each feature, from the number of features within each LSN. Object classes are accumulated in parallel with features, and are used to provide the context for the feature. An associative memory is used to learn the correct threshold for each feature in each context. This use of feature-specific thresholding is novel within the context of the GHT. The principle is illustrated in figure 2. The object is represented by the line features (outlined by solid boxes), and these interact through their LSN (outlined by dotted boxes). Only those features which fall with the LSN will influence another feature directly.

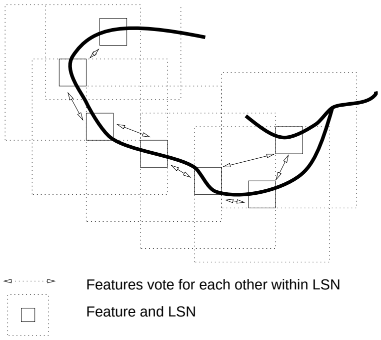

Figure 2: Principle of Local Support Neighbourhoods

This process gives an initial labelling to the features in the image. This labelling of features is then treated as a set of features, each of which votes for other features within the LSN. This information is used to modify the labels assigned to features (in effect , by propagation of labels between local support neighbourhoods), and the process continues until a stable, consistent labelling of the features is obtained. Updates of labels in the accumulator are performed synchronously. Stability is determined by comparison of label states before and after update. Local consistency is assured by the interaction of features within the LSN. Globally, there may be any number of pockets of consistency each of which corresponds to an object or part of an object.

3 Experiments

This section describes the experiments used to examine the effects of the modifications on the performance of the recogniser, compared to an implementation of the GHT algorithm.

For testing purposes, synthetic images have been generated from components of original fax images. Objects to be detected have been added to (generally noisy) backgrounds, and then further noise added on top of the image. The locations of the objects in the synthetic images are recorded so that detection of the objects at the correct position may be automatically verified. Four different types of noise typical of fax images were added to the images at varying levels: (a) random noise – pixels are randomly set to black in the test image, (b) line-blanking noise – pixels are randomly selected as noise pixels, and any line containing noise is blanked from the point of error, (c) line-dropout noise – pixels are randomly selected as noise pixels, and any line containing noise is removed from the image, and (d) morphological noise – black pixels are randomly selected as noise pixels, and a morphological dilation operator applied.

For each test image analysed, an image of the detected features and corresponding object classes is built up. Groups of consistent features represent objects (or parts of objects). When the size of such a group relative to the size of the object exceeds a threshold, and the object is at the correct position, a detection is counted.

3.1 Initial GHT implementation results

The GHT has been implemented as described in [13], and recognition performance has been tested against a portfolio of synthetic test images generated as described above. The results are shown in figure 3. Each point is the result of fifty tests on different combinations of image and noise. For each class of noise, the recognition rate is plotted as a function of the additive noise density. Even with no added noise, the recognition results are not perfect. This can in part be explained by the image synthesis process, which adds the test object to a noisy background and therefore makes recognition more difficult, and in part by the effects of sub-sampling the image features and the effect of sampling alignment on spreading votes through the accumulator. Additionally, where more than one object is present in the image, the thresholding process will return only the largest object.

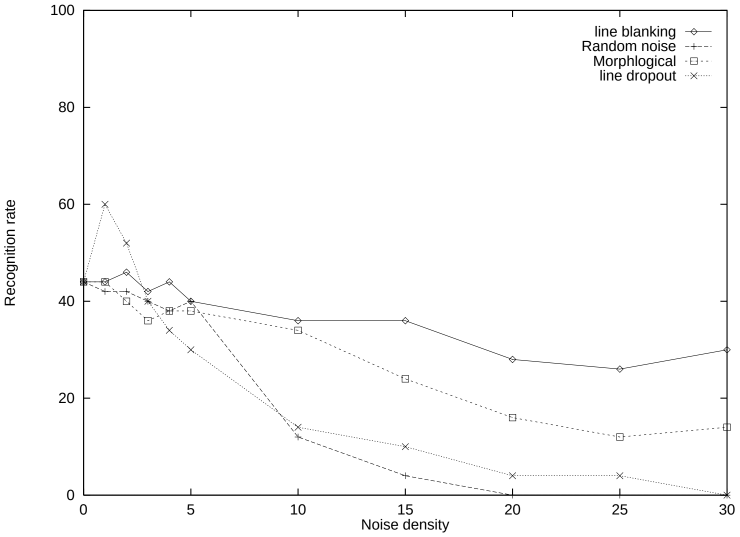

Figure 3: Variation of recognition rate with noise density (GHT system)

As the amount of noise added to the image is increased, the recognition rate decreases as expected. The different types of noise have a greater or lesser effect

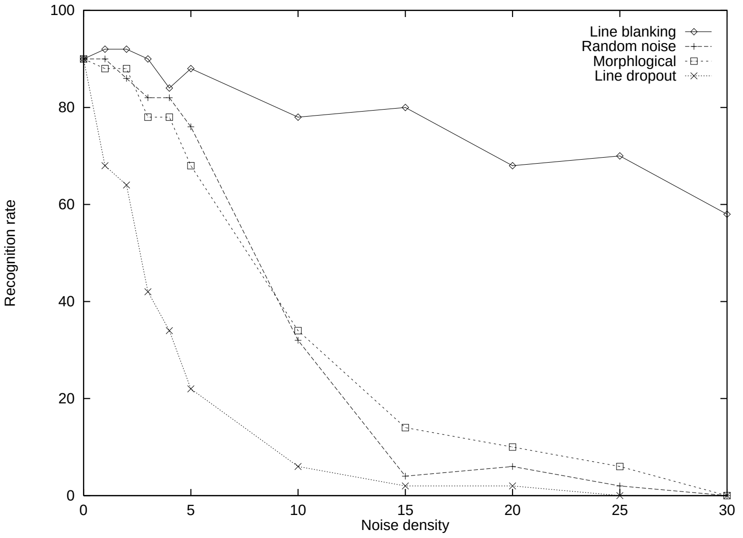

Figure 4: Variation of recognition rate with noise density (LSN system)

on the recognition rate. Unsurprisingly, the greatest effect is from line-dropout noise, which distorts the image. The greatest tolerance is shown for line-blanking noise, where a line with an error is blanked out from the point of error. At the zero noise level, the false positive rate for the image set was about five per image.

3.2 Modified GHT results (LSN)

The addition of the feature-specific thresholding had the desired effect of making possible the detection of multiple classes of objects with varying number so features. Whereas before, the original thresholding process only returned the largest objects, use of the modified thresholding enabled the location of all object in the image

Recognition performance of the system with the addition of the local support neighbourhoods has been tested using the same portfolio of test images as was used in testing the original system, with the same schedule for addition of noise to the image. The results of the experiments are shown in figure 4.

As can be seen, the performance of the system on images with no added noise is much higher that in the original system. This can be explained by the ability of the system to locate and identify sub-objects with confidence and use this information to assist in identifying the rest of the object. As before, the addition of noise in various amounts degrades the performance of the system, and the effects of the different types of noise are broadly similar to the those described in the previous section. Table 1 summarises the noise performance in terms of the level of added noise required to reduce the recognition rate to twenty percent. As can be seen from the figures, the increased performance at low noise levels has been traded at the expense of no increase in robustness at increased noise levels. The false positive rate has however been reduced to about 0.05 per image, a significant reduction.

Noise source	GHT	LSN
Random	9%	12%
Line Blanking	> 30%	>> 30%
Line Dropout	8%	6%
Morphological	18%	13%

Table 1: Noise levels which decay recognition rate to 20%

3.3 Effect of neighbourhood size

The size of the local neighbourhood affects the ability of the recognition system to reject noise. With the minimum neighbourhood size, each feature in the image is treated independently, and there is no gain in recognition rate. As the size of the neighbourhood is increased, each feature is influenced by those features surrounding it, and feature labels which are inconsistent with the rest of the object are detected and modified. With further increases in neighbourhood size, the amount of this contextual evidence increases. However, as the neighbourhood expands the likelihood of inclusion of non-object features increases. These features will tend to dilute the effect of the context information. Thus, we would expect the recognition rate to increase with neighbourhood size rapidly at first, and then level off.

Figure 5 shows the results of experiments in which the system was trained to recognise a set of objects. The system was retrained a number of times, using a different neighbourhood size each time. The neighbourhoods used are square, and the size of the neighbourhood is measure in terms of the number of features over which the influence extends away from the centre. Thus, a neighbourhood of size five may be envisioned as a square of side eleven, centred on the feature (the use of square neighbourhoods is for convenience, and not a theoretical restriction). The recognition rate for the system was measured on a set of synthetic test images with a fixed amount of morphological noise (5%). As expected, the recognition rate does initially increase as the neighbourhood size increases, followed by a levelling-off. The maximum recognition rate appears to occur for a neighbourhood size of 5, although for this relatively small test set, random variations may mask further increases in recognition rate. From a computational perspective, the optimum grid size is selected by trading recognition rate for speed. The computation required increases as a function of the neighbourhood size, making larger neighbourhoods unattractive.

4 Conclusions and summary

We have described a new approach to the problem of identification of complex objects in document images. The novelty of the approach lies in its use of a

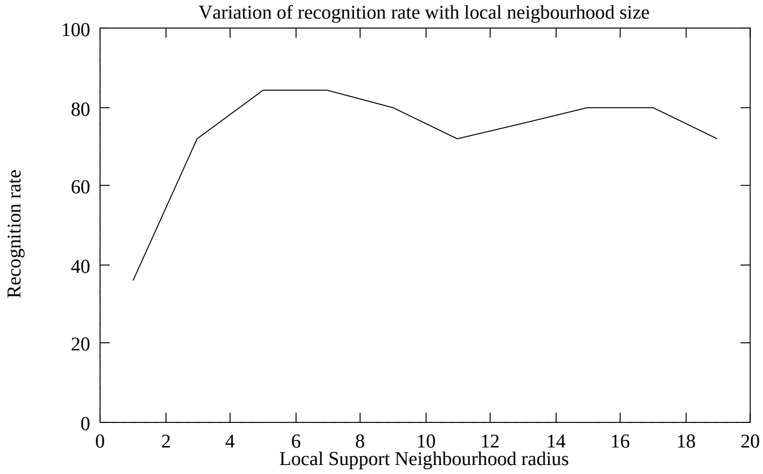

Figure 5: Variation of recognition rate with local neighbourhood size (LSN system)

distributed representation for objects which seeks to overcome some of the problems of data noise and incompleteness. Objects are modelled as a set of features and the relationship between neighbouring features. Each feature in the image stimulates recall from an associative memory of the other features within its local neighbourhood. Feature labels are accumulated locally, and the most likely label is assigned to each feature. The process then updates the labels synchronously until a stable, locally consistent labelling of the image features is obtained. Thus, incomplete image data may lead to recognition and recall of a complete object in translation-invariant, robust manner.

References

[1] Allan Hanson and Edward Riseman. Processing cones: A computational structure for image analysis. In S. Tanimoto and A. Klinger, editors, *Structured Computer Vision*, chapter 4, pages 101–131. Academic Press, 1st edition, 1980.

[2] C.C. Weems, S.P. Levitan, A.R. Hanson, E.M. Riseman, D.B. Shu, and J.G. Nash. The image understanding architecture. *International Journal of Computer Vision*, 2:251–282, 1989.

[3] M. Viswanathan and G. Nagy. Charactersitics of digitized images of technical articles. In *S.P.I.E. Vol. 1661*, pages 6–17. S.P.I.E., 1992.

[4] G. Nagy. Document analysis and optical character recognition. In *Proceedings of 5th International Conference on Image Analysis*, pages 511–529, 1990.

[5] H.S. Baird. The skew angle of printed documents. In *Proceedings of the SPSE 40th Conference and Symposium on Hybrid Imaging Systems*, 1987.

[6] H.S. Baird. Feature identification for hybrid structural/statistical pattern classification. *Computer Vision, Graphics, and Image Processing*, 42 A03(3):318–33, Jun 1988. Comput. Vis. Graph. Image Process. (USA).

[7] H. S. Baird. Document image defect models. In H. S. Baird, H. Bunke, and K. Yamamoto, editors, *Structured Document Image Analysis.* Springer-Verlag, 1992.

[8] S. Shapiro, G. Gluhchev, and V. Sgurev. Handwritten document image segmentation and analysis. *Pattern Recognition Letters*, 14:71–78, 1993.

[9] M. Krishnamoorthy, G. Nagy, S. Seth, and M. Viswanathan. Syntactic segmentation and labelling of digitzed pages from technical journals. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(7):737–747, July 1993.

[10] Suichi Tsujimoto and Haruo Asada. Understanding multi-articled documents. In *10th International Conference on Pattern Recognition.* IEEE, IEEE Computer Society Press, 1990.

[11] F. Fein and F. Hones. Model-based strategy for document image analysis. In *S.P.I.E. Vol. 1661*, pages 247–256. S.P.I.E, 1992.

[12] T. Watanabe, Q. Luo, and N. Sugie. Structure recognition methods for various types of document. *Machine Vision and Applications*, 6:163–176, 1993.

[13] S. O'Keefe and J. Austin. Application of an associative memory to the analysis of document fax images. In E. R. Hancock, editor, *Proceeding of the British Machine Vision Conference*, volume 1, pages 315–326. British Machine Vision Association, BMVA Press, 1994.

[14] S. O'Keefe and J. Austin. Image object labelling and classification using and associative memory. In *IEE Fifth International Conference on Image Processing and its Applications*, pages 286–290. Institution of Electrical Engineers, London, July 1995.

[15] S. O'Keefe and J. Austin. An application of the adam associative memory to the analysis of document images. In D. L. Bisset, editor, *Proceedings of the Weightless Neural Network Conference - Computing with Logical Neurons*, pages 17 – 22. University of Kent at Canterbury, September 1995.

[16] S. O'Keefe and J. Austin. Image labelling using an associative memory. In *Proceedings, ICANN'95 - International Conference on Artificial Neural Networks*, pages 281 – 286. European Neural Network Society, EC2 & cie, Paris, October 1995.

[17] D. H. Ballard. Generalising the hough transform to detect arbitrary shapes. *Pattern Recognition*, 12:111–122, 1981.

[18] V. Leavers. Which hough transform? *CVGIP: Image Understanding*, 58(2):250–264, September 1993.

[19] W.E.L. Grimson and D.P. Huttenlocher. On the sensitivity of the hough transform for object recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 12(3):255–274, 1990.

[20] M. Yamada and K. Hasuike. Document image processing based on enhanced border following algorithm. *Proceedings 10th IAPR Conference*, pages 231–236, 1990.