DICTA-91 Digital Image Computing: Techniques and Applications
Royal Exhibition Building Conference Centre, Melbourne 4-6 December 1991
ed. A.J. Maeder and B.M. Jenkins *Australian Pattern Recognition Society* ISBN 0 646 07338 9

ACQUIRING INFORMATION FROM CUES

Paul L. Rosin

Cognitive Systems Group
School of Computing Science
Curtin University of Technology
Perth, 6001
Western Australia
email: rosin@cutmcvax.cs.curtin.edu.au

ABSTRACT

A standard technique in computer vision is the use of low-level image cues to direct high-level reasoning. This paper argues that previous use of cues has been inadequate on two counts. First, much of the available information relating to an image feature is often not acquired. Second, that the extracted information is not fully utilised. Several suggestions are made regarding better use of cues, and are demonstrated by examples.

1. INTRODUCTION

Humans use cues from their five senses to aid the interpretation of their environment. Cues are simple features that act as triggers or guides for perception or action. They can be considered as information rich selections of low-level data passed on to higher levels. In a similar manner, computer vision systems can use cues from the various modalities to aid image analysis. Cues are generally formed bottom-up (in the case of humans they may be preattentive) without the need for model-based knowledge. Since they are derived from inexact and uncertain data, they are only suggestive, providing hypotheses for top-down reasoning. To improve their effectiveness different classes of cues are often used in combination.

A bottom-up cue-based strategy can provide a number of benefits for a computer vision system. One example is model selection. Most early computer visions systems were restricted to constrained environments and only modelled a very small number of objects. In such circumstances a simple approach to recognition is an entirely top-down strategy. In its simplest form this involves selecting each model in turn and matching it to the scene. However, for systems to emulate the performance of the human visual system they must be able to recognise hundreds or thousands of objects. If large numbers of object models are stored then recognition cannot be performed rapidly if every model has to be considered individually. An alternative and more efficient approach is to derive a set of cues from the image. Model selection can then be performed bottom-up if each cue indexes either one or a small set of likely models which can then be matched to the image.

In the past researchers have used a variety of cues for computer vision. However, their power is generally not used to the full. Much of the available useful information present in cues is either not acquired or is discarded. An example of an under-utilised cue is the corner. Corner detectors usually return just a single value measuring the strength of the corner. However, a corner contains much more information such as its orientation, the subtended angle, how sharp or blunt (rounded) it is, and its size (i.e. over what scale the corner exists).

All this additional information can be put to good use by more fully restricting the possible interpretations of the image. In turn this can improve the accuracy and efficiency of the succeeding stages of processing. First, because simpler techniques can be used when possible, and second, because techniques made more powerful by increased specialisation or fine tuning to the image become applicable. This paper lists some of the classes of information that can be determined from cues, and describes how they can be used to improve image analysis. Some of these are demonstrated by examples in a computer vision system called FABIUS.

2. INFORMATION FROM CUES

There are a number of criteria for a good cue: robustness to occlusion, background clutter, and noise; invariance over a range of viewpoints; specificity; and computational efficiency.

Both robustness and invariance are important so that the cue can be reliably extracted under various conditions. A cue that could only be extracted from very clean images or was restricted to one particular view of the object would be of little general use. An example of cues satisfying these two conditions are the line grouping rules based on non-accidentalness [4]. For instance, two parallel lines in a scene would be declared a significant cue only if their spatial arrangement was unlikely to arise by the chance positioning of unrelated lines.

Depending on the type of cue different amounts of information can be acquired directly from the scene. Typically, the more complex the cue, the more powerful it is; i.e. it provides more information and the possible interpretations are more restricted. On the other hand, complex cues have several disadvantages over simpler cues. They are more computationally expensive to generate. They will be more specific so that separate cues will be needed even for similar models. In addition, they will be less robust to occlusion and noise.

Within vision cues can take many forms and provide different sorts of information. A common class of cues are those that infer three dimensional structure of portions of the scene from one or several images. This process is often called *inverse optics* or *shape from X*, and can be derived from stereo, motion, parallax, perspective, convergence, texture, shading, colour, shadows, and contour. 3D inferences can also be made from other features such as arrangements of straight and curved lines satisfying general perceptual grouping rules such as parallelism, colinearity, and proximity [4], and groupings of ellipses [9]. Other examples of cues are: corners and vertices; axes of local grey-level symmetry [12], regions with shape attributes such as symmetry or compactness, or interior attributes such as homogeneity or high/low intensity; circular arcs [10]; combinations of lines such as S [2] and U [6] shapes; closed polygons [2]; and groupings of other features such as blobs, and other lower-level groupings themselves [5]. The different types of information that can be derived from various cues are discussed below.

2.1 Model Selection

Cues may indicate either a simple model or a set of possible models. There is evidence that humans represent models as specialisation hierarchies and that objects are initially identified at a certain *basic level* within the hierarchy [7]. Likewise, computer vision systems can use cues to select models at any level from the specific to the general. The appropriate level of specialisation to index a model will depend on the available *a priori* knowledge, the set of models, and the scene. These factors also affect the reliability and efficacy of the cues [1]. An example of generic cueing is given by Fretwell *et al.* [3] who use the wiggliness of curves to distinguish between only two general classes of objects: bushy vegetation and man-made objects. At the opposite extreme, Lowe's SCERPO system [4] uses cues based on perceptual groupings of lines to indicate not just the model, but also its approximate pose.

2.2 Location

Cues are often used to indicate the location of the model. There need not be a direct correspondence between the cue and a specific part of the model in which case the cue would only indicate an approximate location within a window centred on the cue. An example of this is given by the generic cueing described above [3] which does not specify a model and therefore cannot locate the position of the object of interest with any exactness. If there is a direct correspondence between cue and model this will provide a partial match of the model to the data, determining a model pose (i.e. translation and rotation). Of course, there may be several possible partial matches, or the match may not restrict the model to a unique pose, in which case the cue provides several possible poses. For example, if a circle is used as a cue then a match to a single cue will specify the set of poses with a rotational symmetry about the centre of the circle.

2.3 Scale

A cue may also provide scale information. Examples of such cues are parallelism (scale determined by the separation between the lines), circular arcs (scale determined by arc radius), U and S shaped polylines (scale principally determined by the middle line which is the only one guaranteed complete), and closed polygons. Determining scale is extremely useful as it can be used for a variety of functions. First, knowing the scale simplifies matching. If translation and rotation have also been determined then the cue has determined the model identity and its complete pose and appearance. All that requires to be done is to verify the hypothesis by matching the rest of the model to the image.

A second use for the scale is to direct any further feature extraction to be used for matching the model. Since in the real world structures appear at many different scales their size in the image is not known beforehand in unconstrained environments. This leads to problems since it is not possible to design a feature extractor that is optimal at all scales. Two solutions are often taken: either the image is analysed at many scales, which is computationally expensive; or feature detection is applied at one scale only to give sub-optimal but reasonable performance for different sized structures. However, an alternative is possible if an object's scale is determined when analysing the image from a cue. Then features can be extracted at the appropriate scale from the window covering the expected location of the object in the image. As well as improving effectiveness, concentrating on a region of interest also improves efficiency since only a small portion of the image needs to be processed further. This same strategy is used by humans who by focussing their attention on the small areas in a scene containing high amounts of information improve their response to fine detail [11].

Another use of scale is to specify a technique for matching the model and image data. The most appropriate matching procedure will depend on several factors such as which model is being matched, the type of scene, the results of prior processing, and scale. For instance, small instances of an object will contain much less detail and will be more prone to noise and occlusion than larger instances. On the other hand, objects seen at a high scale may contain excess detail that obscures the features of interest. Therefore, objects at different scales are best matched using different methods.

In addition to selecting the type of matching to use, the parameters to the matching procedure can be determined by the estimated scale. For instance, because the features of smaller objects will be less accurately extracted, the thresholds on the matcher may need to be lowered if the small objects are to be detected.

2.4 Reliability

Finally, when extracting a cue information may also be obtained describing the reliability of the cue detection. The reliability measure will depend on the feature extraction technique and the relevant feature. For example, when detecting circles the reliability could be based on the spread and size of peaks in accumulator space if the Hough Transform was used, the residual error if least-square fitting was applied, or some function of the pointwise error. The cues may then be ranked so that the most reliably detected ones are evaluated first.

The reliability of cue extraction may also be taken as an indication of local noise and clutter in the image. Any following processing may then be guided by this knowledge. If the image is fairly free from noise and clutter then a simple and fast processing technique may be sufficient. Otherwise more complex techniques will be required. Likewise, the other parameters to the routines for feature extraction may be determined by the estimated noise. For instance, thresholds can either be raised to exclude noisy clutter, or they can be lower to ensure that the features of interest are detected.

3. AN IMPLEMENTATION OF CUEING

The extraction and application of cues has been implemented within a computer vision system called FABIUS (see reference [10] for a full description of FABIUS). All the models and much of the data is stored in FABIUS as hierarchies of frames. The frame representation provides a great deal of flexibility, allowing the cue-based information to direct processing. This is designed to complement FABIUS's "toolbox" philosophy. A range of procedures is expected to be available to perform any

task (such as feature extraction, model invocation, and model matching). Although this involves some redundancy, each procedure will have its own advantages and disadvantages, and will perform better than the other procedures in certain circumstances. The vision system can thereby gain a high performance by selecting the most appropriate procedure from the toolbox for the task at hand. Which tool is selected will depend on factors such as the problem domain, control strategy, complexity of the image, state of interpretation of the image, and the hypothesised model. We show here how the information derived from cues will be used to guide the tool selection operation.

One of the applications of FABIUS has been to recognise side-on views of vehicles in outdoor scenes. Examples will be taken from this application to demonstrate the use of cue-based information. The use of cues to select models is already well established and will not be further pursued here. For simplicity examples will be restricted to the matching of a single model: a bicycle. The bicycle model is hierarchical, and contains two main components, the wheel pair and the frame. The wheel pair is specified as two circles of similar size separated by a distance dependent on the wheel size. The frame consists of six straight line segments positioned relative to the wheel pair. The appearance of the two components of the bicycle in the image have different characteristics. The detection of the wheels is generally straightforward and robust. Detecting the frame detection is problematic since it contains relatively fine detail, and is often obscured by the rider's legs. That is why circles were chosen as the most suitable cues for the bicycle, where a circle indicates one of the wheels. The following examples all involve the use of circles as cues and describe the available information that can be extracted from them. In particular it is shown how the circle cues can compensate for the fragmentation and incompleteness of the appearance of the frame by guiding the stages leading to the matching of the frame.

The examples are intended to demonstrate the ideas of cueing rather than present a rigourous theory or procedure. To that end, the decisions based on cue values have been made by qualitative rules acquired experimentally rather than as functions of precise combinations of values.

3.1 Cue Formation

Figure 1 shows the original image. This is edge detected, and adjacent edge pixels are linked into lists. Only lists whose average edge strength is above a threshold are kept, and are approximated by straight lines and circular arcs (drawn in bold) [8] as shown in figure 2. Arcs are hypothesised as arising from circles. Circles with close centres and similar radii are combined. All circles whose radii lie within a certain range are then proposed as cues. In this application smaller or larger circles can be rejected since the corresponding model (the bicycle) would be either so small or large that it could not be recognised in the image due to inadequate resolution or lack of model coverage. Each cue has two attributes in addition to its radius and centre location: 1) the proportion of the circle covered by the underlying arcs, and 2) the goodness of fit of the arcs - this is the significance measure used by the segmentation program [8]. The circle cues generated from the arcs in figure 2 are shown in figure 3. Since arc detection is robust, and the formed circles are only intended to be suggestive, the parameters to the initial edge detection procedure are not critical. (The segmentation procedure for forming arcs does not require any parameters.)

3.2 Model Matching

A match of a bicycle wheel to a single circle cue restricts the pose of the model to a rotational symmetry about the circle - the scale of the model is uniquely determined by the radius of the circle. For each cue a second matching circle is sought to complete the bicycle wheel pair model. Once a valid match is made to the second wheel the pose is further restricted to four positions: the bicycle faces either left or right, and is upright or upside down. Each hypothesised wheel pair is verified by matching the second component of the model - the frame. The standard matching technique used is a template match. The model is projected onto the image and nearby lines with similar orientations to the model are accumulated. This accommodates the fragmentation of the frame as it allows many-to-one matches to be made. If the overlap between the image lines and the model is above a threshold then the match is accepted. Figure 4 shows an example of an incorrect wheel pair hypothesis and the most successful bicycle match. The bold lines are image data matched to the model. To allow for occlusion the threshold for a successful match has been set low. An overlay of the matched model on the original image is shown in figure 5.

3.3 Dealing with Noise

To demonstrate the power of cues the original image is degraded by adding noise (see figure 6). When processed as before the extracted lines and arcs (shown in figure 7) are noisier, as would be expected. In addition, a large number of low contrast edges have been extracted. The extracted circle cues are shown in figure 8 and it can be seen that the circles corresponding to the bicycle wheels have still been successfully extracted - verifying their robustness. However, due to the underlying data being noisier than in the original example the arcs that form the circle cues have been fragmented and have a poorer significance (i.e. goodness of fit), indicating that the image is noisier than before. Since the frame is susceptible to noise and clutter, this knowledge is used to remove some of the effects of noise. The cue has also localised the model, and within this window the image is smoothed by a 3x3 averaging mask. Furthermore, the edge pixel linking stage is run with a higher threshold to remove some of the spurious low contrast edges. For comparison the original edge data is shown in figure 9, the noisy edge data in figure 10, and the cleaned noisy edge data in figure 11. It can be seen that much of the effects of noise have been successfully removed. The edge data is approximated by straight lines as before, and the frame is matched. The best match is shown in figure 12 and comparison to a match based on the unimproved data (figure 13) shows a distinct improvement. Although even the improved match is still very sparse this is to be expected in a noisy image. The threshold for a successful match has been appropriately lowered based on the circle cue's noise estimate.

3.4 Adapting to Different Scales

An alternative method of degrading the data is given by subsampling the 512x512 image to form a 256x256 image. Again the extracted arcs (figure 14) successfully generate the cues corresponding to the bicycle (figure 15). However, due to its fine detail practically all of the frame has been lost. This is overcome by using the scale information in the circle cue to suggest that the window of interest be edge-detected again with parameters set for finer detail. It can be seen from the results of the second edge detection pass (figure 16) that the bicycle frame has been much better extracted than the original edge detection (figure 17). The best frame match is shown in figure 18. Again, based on the circle cue's scale information the threshold for a successful match has been lowered.

3.5 Selecting Appropriate Procedures

The final example shows the lines extracted from a high quality image of a bicycle with little noise or clutter (figure 19). The circle cues are based on almost complete arcs with good fits, indicating a very clear instance of the bicycle. This knowledge is used to select an alternative matching technique to the template matcher. A simple search in the image for lines matching the frame successfully locates the bicycle frame. This matching technique is more efficient than accumulating all partial lines matches, but can only be applied to good images containing one-to-one line matches.

4. SUMMARY AND CONCLUSIONS

The use of low-level cues to direct high-level reasoning is commonplace in computer vision. However, many cues provide much more information than is currently used. This paper listed some of the classes of information that can be derived from cues such as model identity, location, and scale, and estimates of image noise and clutter. A computer vision system called FABIUS was used to demonstrate a series of examples in which circle cues were extracted. The cues were used to determine the information detailed above, which in turn was used to: guide feature extraction; select the most appropriate matching technique; and set thresholds for acceptable model matching. It was shown that by using the additional information present in cues the performance of the computer vision system was improved.

5. REFERENCES

1. Biederman I., Mezzanotte R.J., Rabinowitz J.C., "Scene Perception: Detecting and Judging Objects undergoing Relational Violations", Cognitive Psychology, Vol. 14, pp. 143-177, 1982.

2. Bodington R., Sullivan G.D., Baker K.D., "The Consistent Labelling of Image Features using an ATMS", Proceedings 4th Alvey Vision Conference, Manchester, U.K., pp. 7-12, 1988.

3. Fretwell P., Bayliss D.A., Radford C.J., Series R.W., "Generic Cueing in Image Understanding", Lecture Notes in Computer Science, Vol. 301, ed. J. Kittler, Springer Verlag, 1988.

4. Lowe D.G., Perceptual Organization and Visual Recognition, Kluwer Academic Publisher, 1985.

5. Marr D., Vision, Freeman, 1982.

6. Mohan R., Nevatia R., "Using Perceptual Organization to Extract 3-D Structures", Transactions IEEE PAMI, Vol. 11, pp. 1121-1139, 1989.

7. Rosch E., Mervis C.B., Gray W.D., Johnson D.M., Boyes-Bream P., "Basic Objects in Natural Categories", Cognitive Psychology, Vol. 8, pp. 382-439, 1976.

8. Rosin P.L., West G.A.W., "Segmentation of Edges into Lines and Arcs", Image and Vision Computing, Vol. 7, pp. 109-114, 1989.

9. Rosin P.L., West G.A.W., "Perceptual Grouping of Circular Arcs under Projection", Proceedings 1st British Machine Vision Conference, Oxford, pp. 379-382, 1990.

10. Rosin P.L., Ellis T.J., "A Frame-Based System for Image Interpretation", Image and Vision Computing, forthcoming 1991.

11. Shulman G.L., Wilson J., "Spatial Frequency and Spatial Attention", Perception, Vol. 16, pp. 103-111, 1987.

12. Thornham A., Taylor C.J., Cooper D.H., "Object Cues for Model Based Image Interpretation", Proceedings 4th Alvey Vision Conference, Manchester, U.K., pp. 53-58, 1988.

figure 1 - original image

figure 2 - extracted lines and arcs (shown bold)

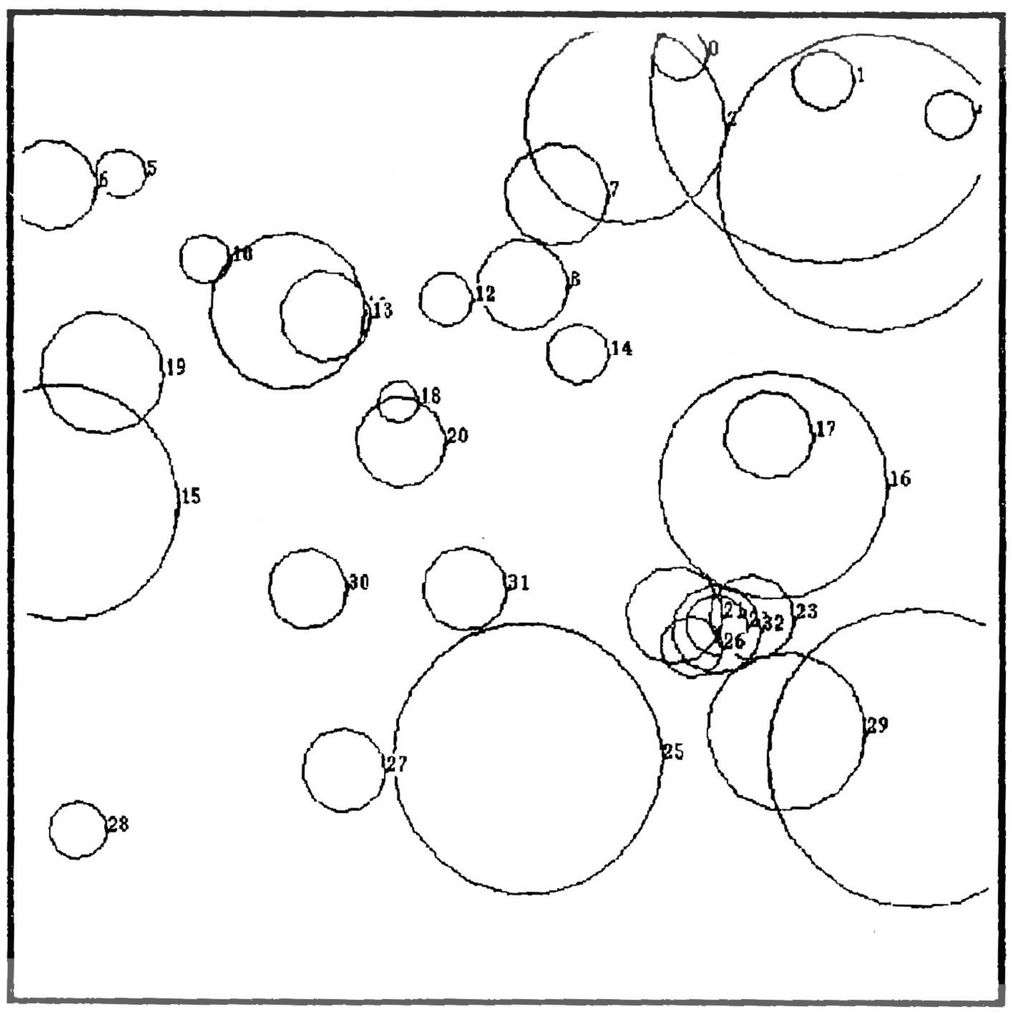

figure 3 - circle cues

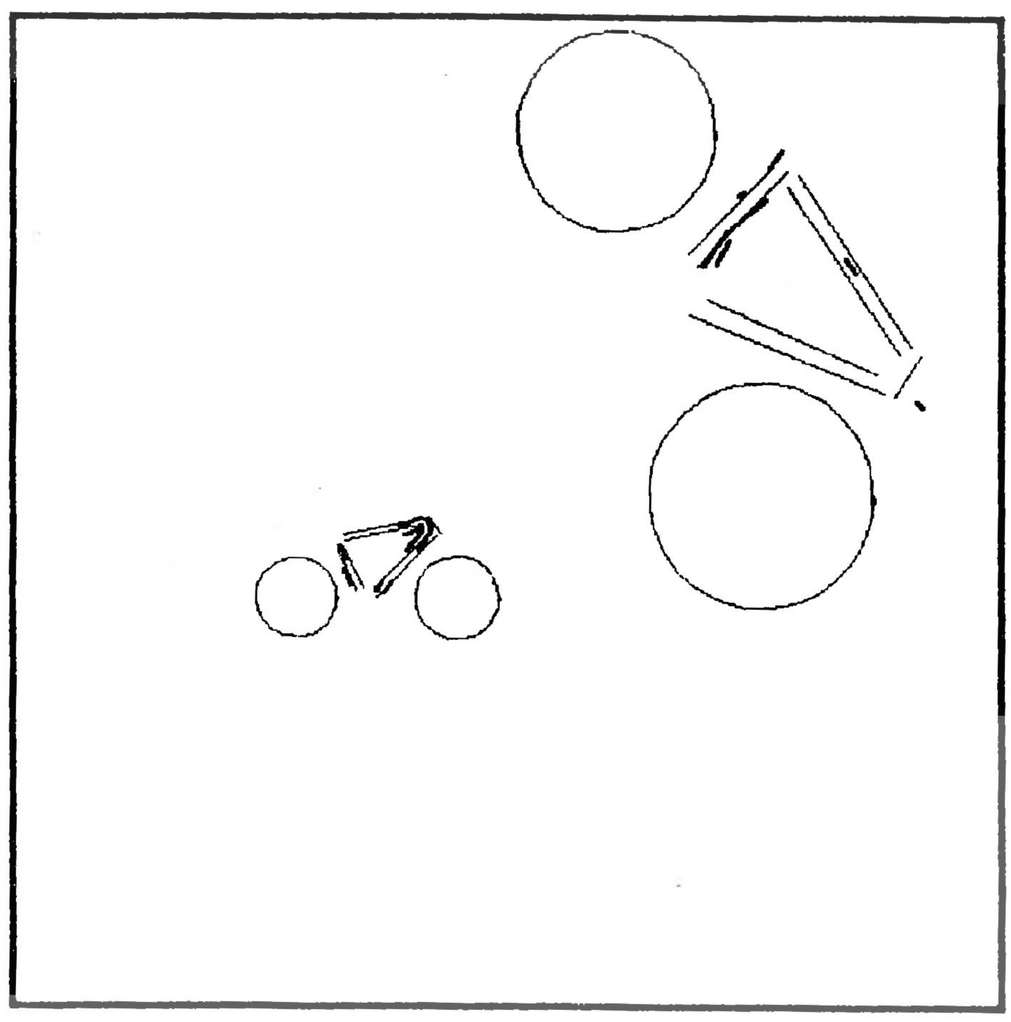

figure 4 - one false match and the successful correct match

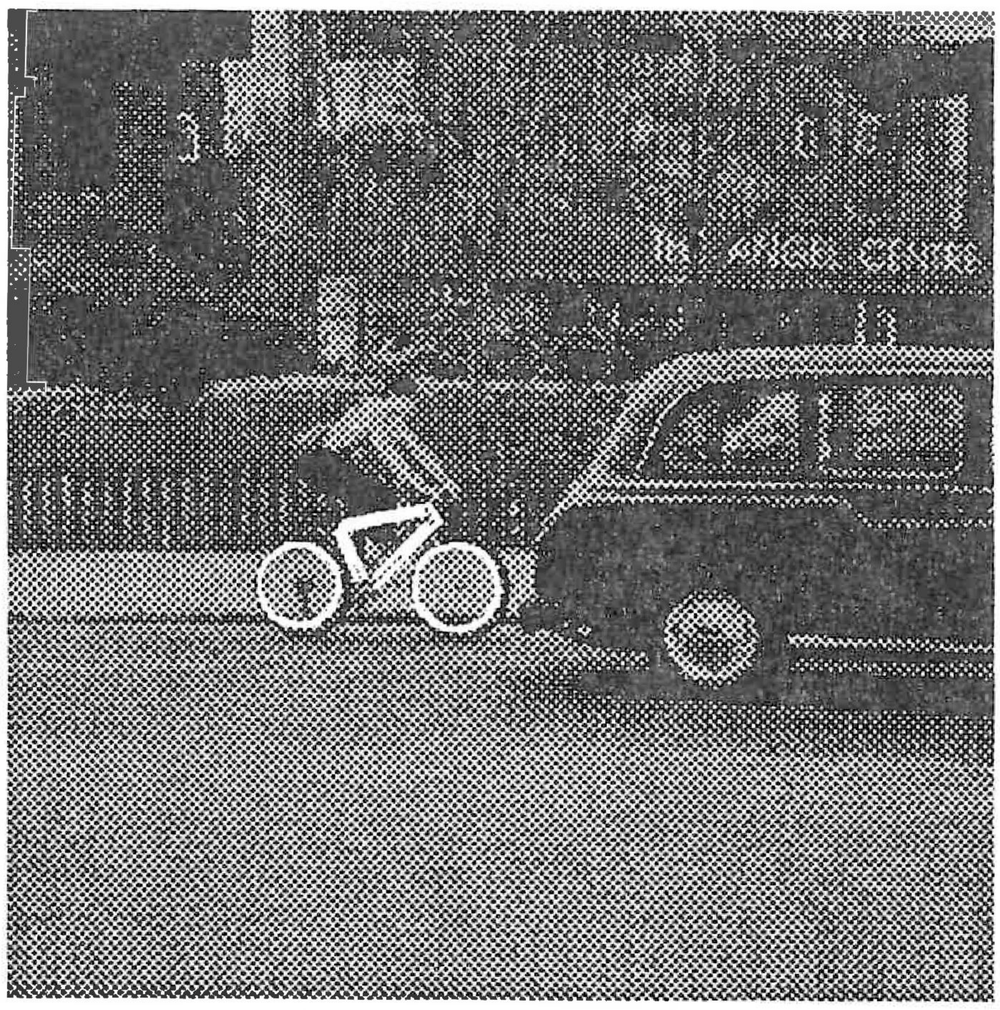

figure 5 - original image overlayed with the successful match

figure 6 - original image with added noise

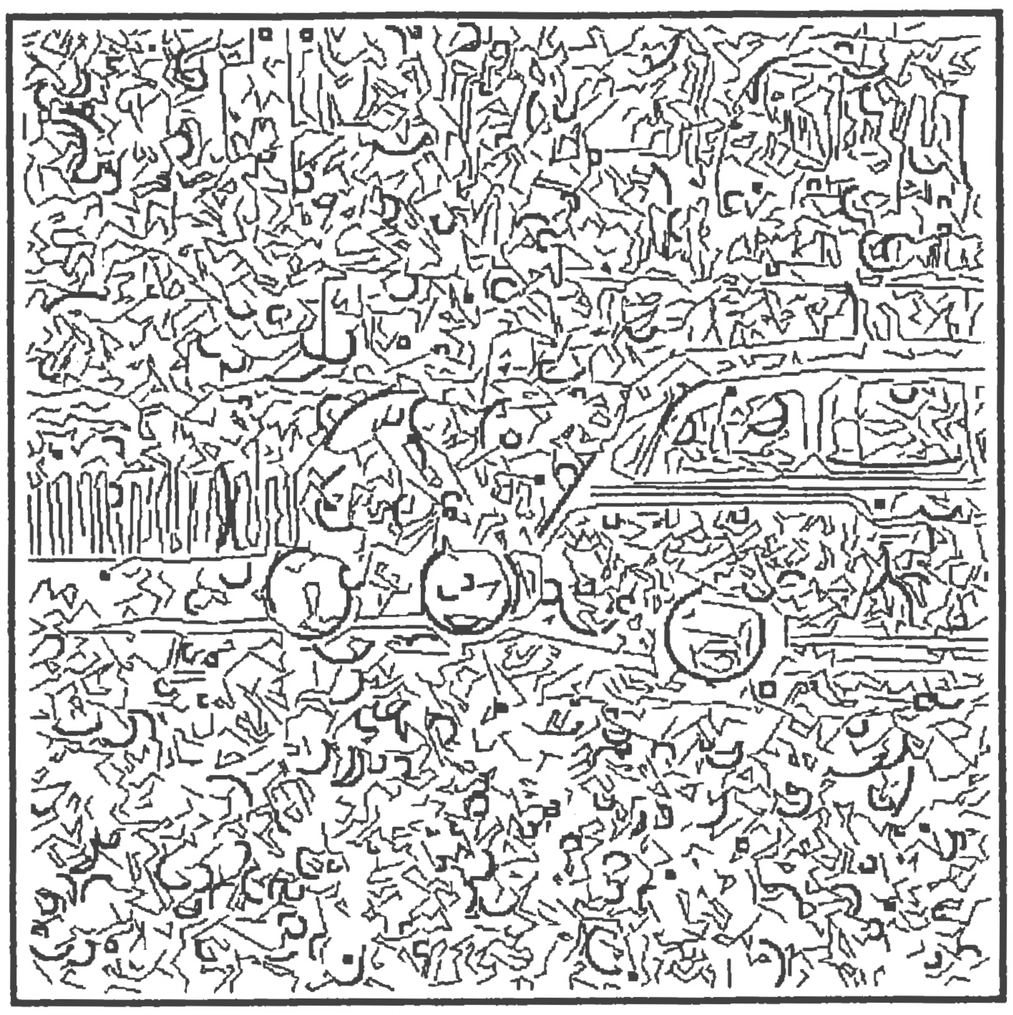

figure 7 - lines and arcs extracted from the noisy image

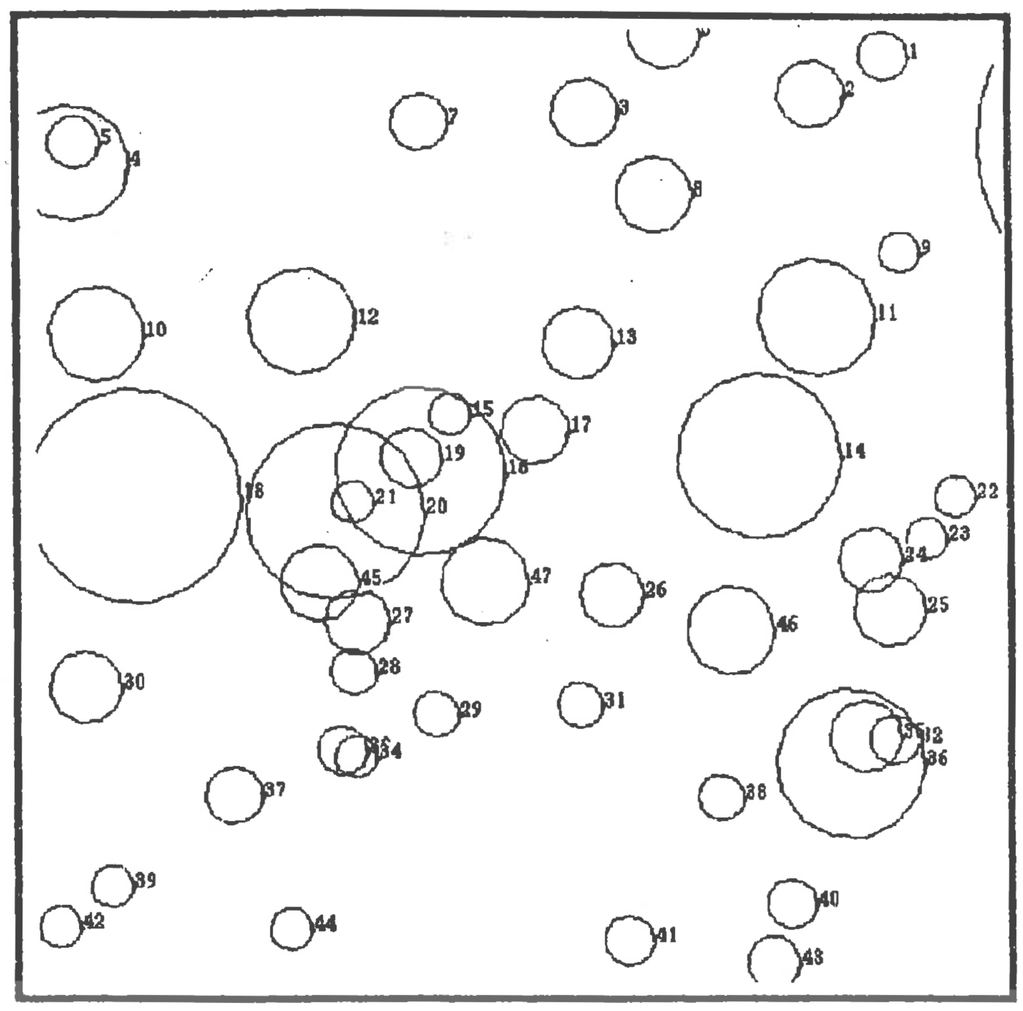

figure 8 - circle cues

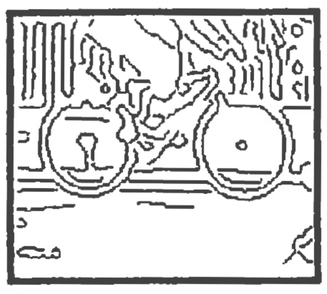

figure 9 - original edge data

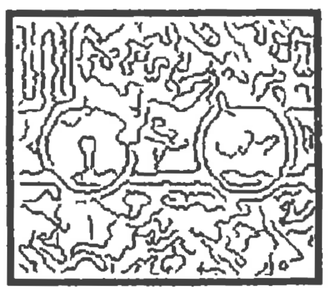

figure 10 - noisy edge data

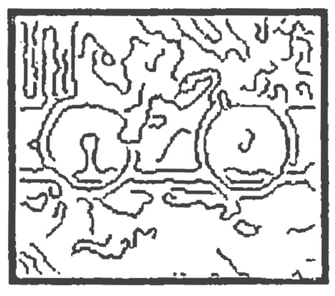

figure 11 - cleaned noisy edge data

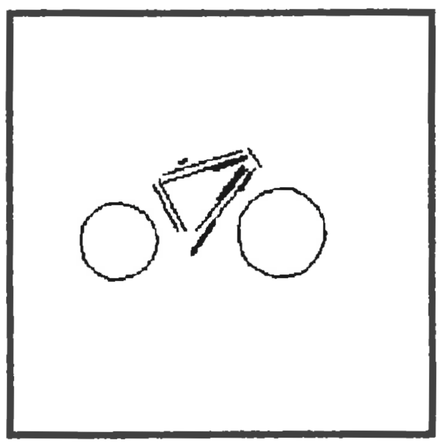

figure 12 - match to clean data

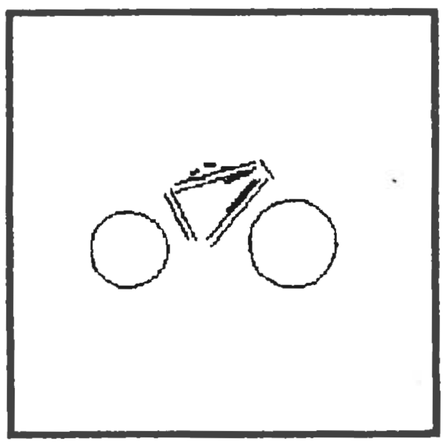

figure 13 - match to noisy data

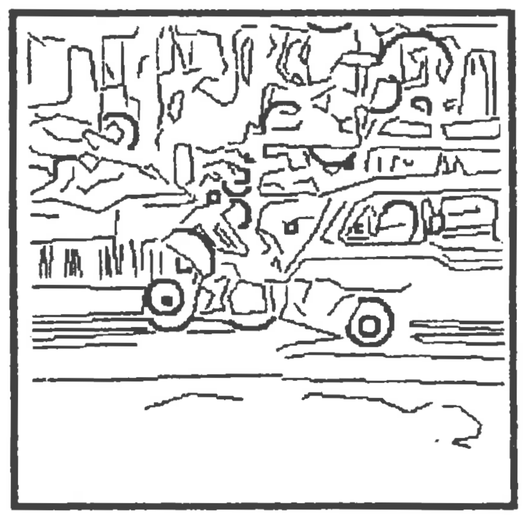

figure 14 - lines and arcs of subsampled image

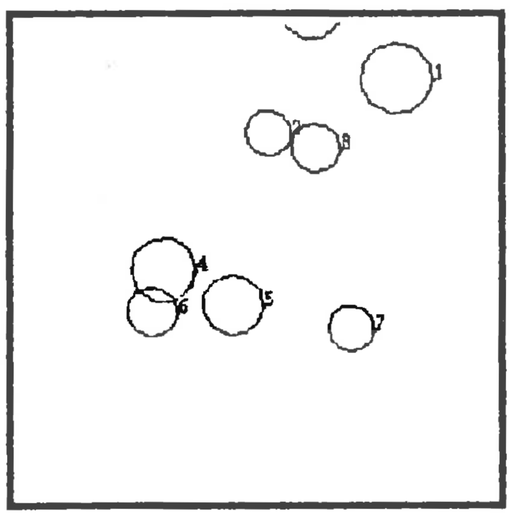

figure 15 - circle cues

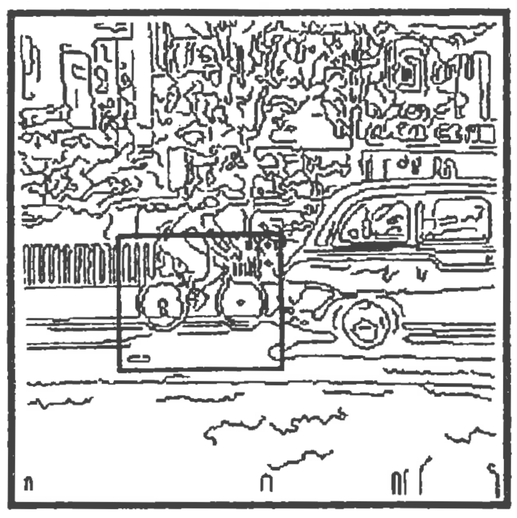

figure 16 - fine edge detection

figure 17 - original edge detection

figure 18 - matched model

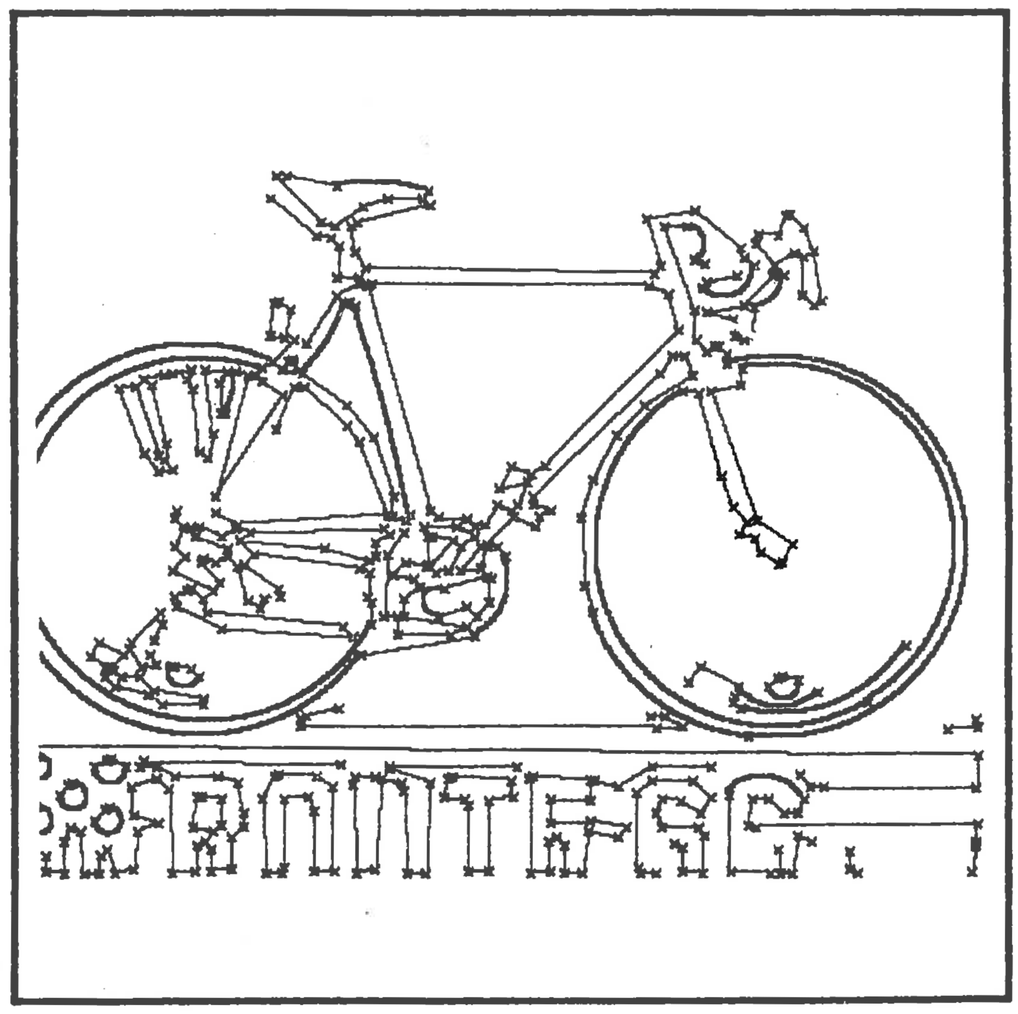

figure 19 - lines and arcs from clean image