SOFTWARE SPEEDUP TECHNIQUES FOR BINARY IMAGE OBJECT RECOGNITION

Alan L Harvey
Communication & Electrical Engineering Department
RMIT Melbourne
3000 Victoria Australia

Harvey A Cohen
Computer Science,Computer Engineering Department
La Trobe University Melbourne
3083 Victoria Australia

ABSTRACT

The use of template matching methods to locate objects in large images is very computationally expensive. This paper describes, for binary images, a coarse to fine technique which speeds up the column and row position search by a factor of 10 or more. A matching error function is used to switch between coarse and fine search modes. Image brightness differences between template and image object will affect matching accuracy and converting template and image to binary format reduces this problem. Also we describe what we term as a sparse template technique and how it is used to give up to a 64 fold speed up or even more for larger templates. This paper reports speed up times obtained on a number plate location and reading project. This work is of importance in vision guided assembly operations where machine vision techniques are used for locating parts. These speedup methods were first tried with grey scale images, [6] and found to be successful. This paper describes the results of using these methods with binary images.

INTRODUCTION

Although most flexible manufacturing cells rely on mechanical parts location systems, the effective use of vision systems has huge potential for truly flexible assembly operations. Flexible manufacturing cells require fast identification of objects for subsequent robotic assembly. Fast image analysis of the camera image data is therefore essential to allow the robot to locate and assemble the parts in the manufacturing cell. This paper describes speed up methods for object location. The application of these speedup methods for objects with unknown orientation and scale is the subject of ongoing study. Objects within an image may be identified by moving templates of the objects to be found over the image until a match is found. If an exhaustive search is made at 'all image positions with templates of significant size considerable computer time will be expended before completing the search. This paper describes methods for speeding up binary image searches for an object within the image using the template matching approach. The software speed up methods described are a coarse fine search technique and a sparse template method. These techniques may be used together to achieve speedup factors of over 60:1. As image brightness is a variable that will affect the matching accuracy it is common practice to convert grey scale images to binary ie black and white images to eliminate brightness as a variable. Selection of a thresholding technique is an important task which is covered in references [4] and [5]. In the coarse-fine search technique a coarse search is made until an indication of an impending match is found. At this point a fine search at every column position is made. The criterion used is the rate of change of the template-image matching error as the template approaches the correct m+atch. Note the figure showing the rate of change of this error for a binary image and template as the template passes through the correct x-y location for the image object. Also note the pictures showing the image and the number plate template.

TEMPLATE IMAGE ERROR MEASURES

The simplest error measure or distance measure is the sum of the absolute image-template differences, ie the Chebyshev error function.

Error = sum|X[i][j]-x[a][b]|

where E is the error sum and X and x are the image and template brightness values. This contrasts with the Euclidean norm, the sum of the errors squared. For integer data, the Chebyshev norm is faster to calculate than the Euclidean norm using contemporary work station architecture. For binary images, however, these two norms are identical, remembering that the image brightness values are equivalent to 0 and 1. An exclusive or operation between the image and template data in fact can be used. On the same data an exclusive or sum took 60 seconds whereas the 16 bit integer difference sum tok 66 seconds, a minor difference of 10%. In this paper, arithmetical differences were used to give a comparison with grey level image calculations. Note however that a 32 bit long integer sum took 78 seconds. For a large template requiring a long integer sum, the exclusive or sum will be some 30% faster. The matching error function is discussed and how it is used to reduce the number of column positions to be searched.

 CH2976-9/91/0000-1827 $1.00

Correlation measures looking for maximums may also be used and reduced searches would apply to correlation also. Template error calculations may be speeded up by the use of sparse or partial templates in which only every say fourth column and row brightness value of the template is checked against the image to be searched. A 16 times reduction in error calculations can be made for this case. Also, the template calculation may be speeded up by the use of a small template. For the case of a square template, the total number of matching calculations T, for a template of size NxN and an image of size MxM is: $T=N*N*(M-N)*(M-N)$. From a consideration of this relationship, the smaller one can make N, the smaller the number of matching calculations required for a full search. See [5].

In the case of searches for discrete objects this is not an option. In the case of finding a position on a textured surface, the template size is a variable and in this case it is desirable to reduce the template size.

TEMPLATE PREPARATION

Firstly, the whole image was thresholded and then a section of the image corresponding to the number plate area was cut out by means of a small program to produce a 96x44 bit array for the template. A global threshold of 105 was used for the car image. Autocorrelation of the template was then done to produce these results. In a practical application, cross correlation rather than autocorrelation would be used.. However these speedup methods would still apply, probably with a reduced speedup factor. See the pictures showing the image and the template.

COARSE FINE COLUMN SEARCH

As a first approach to object location in an image, the bit map to be matched to the image may be moved over the image, column by column and matching calculations made. Clearly if the matching calculations could be made at a reduced number of locations. ie a coarse search at say every fourth or fifth pixel then a large speed up in the matching operation could be made. If a check is made by comparing the current matching error with the error calculation at the previous position then a decision may be made to move along say four or more columns if the error is not moving towards a match. Alternatively, if the matching error is decreasing rapidly, then the bit map would then be moved along column by column. Investigations showed that the bit map may be moved along up to 6 columns for the coarse search and still always find the matching position. A five to one speed up was obtained in this way.

COARSE FINE ROW SEARCH

Similarly, a coarse to fine column search using some change in the image matching criterion between rows will give a speed up factor of the order of four or five to one assuming the spatial changes in the matching function in the vertical direction change in the same way as in the horizontal direction. Clearly this will be the case. However matching error information is not as readily available as it was in the case of the reduced column search. The matching error change method used was to save the lowest value of the matching error for the previous two rows and make a coarse fine search decision on the basis of the difference. An important point to consider in partial searches is the dimensions of the template. A template four columns wide will have a much narrower correlation region than a 40 column wide template. Thus in certain cases for instance whole of image movement, where the template size is not fixed, it may be better to make the template longer (more columns) but not as wide, ie less rows if a coarse column search only is to be made.

SPARSE TEMPLATE MATCHING

Instead of using the entire bit map array to match against the image it is reasonable to use less than the total bit map by employing a sub-grid approach. Template and image brightness values on a 4x4 grid for example would reduce the number of image calculations by a factor of 16. A 16x16 row column grid was used giving a highly significant reduction in template matching calculation time. Clearly this sparse template method would reduce the signal to noise ratio of the matching function as compared with a complete template array calculation so this method should not be used with very noisy images. However it may well be possible to use a multi resolution method here also. This could be achieved by using the change in the image matching function to switch to a full search rather than sub-sampling when there is an indication that the search is close to the vicinity of the object. See [11].

RESULTS

Coarse-Fine Search Technique.

A search for a number plate bit map gave the results shown on the upper diagram, figure 2. The image was that of a car shown in fig 1 and the template consists of the number plate frame of that car. The graph shows a plot of the Chebyshev matching function, the sum of the absolute differences versus the image column number for the correct row number, ie the row number where the template was actually found, row 90. In this diagram a matching calculation was made at every column

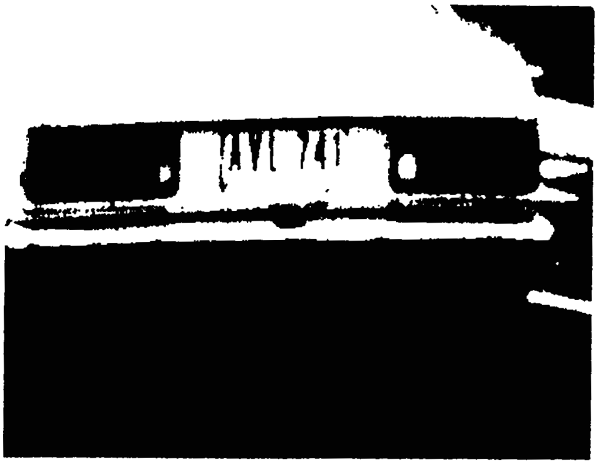

Fig. 1 Car rear view

Fig. 2 Number plate Template

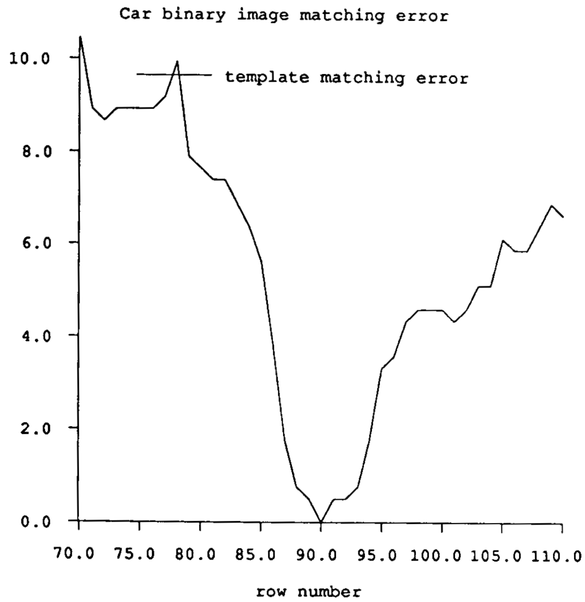

Fig. 3 Matching error down column

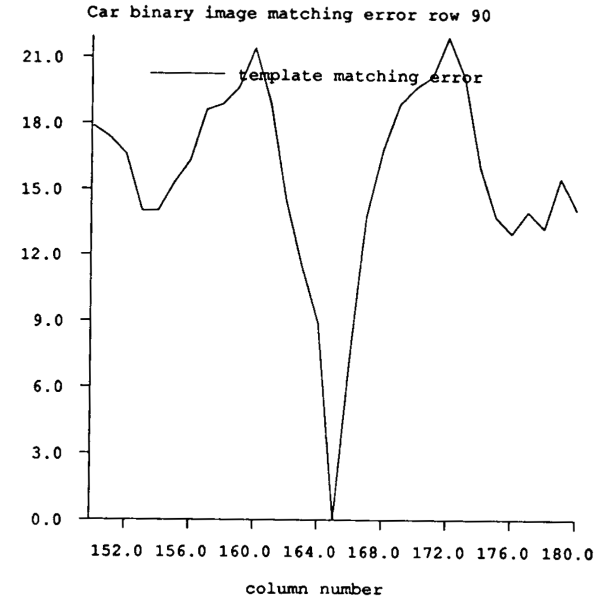

Fig. 4 Matching error along row

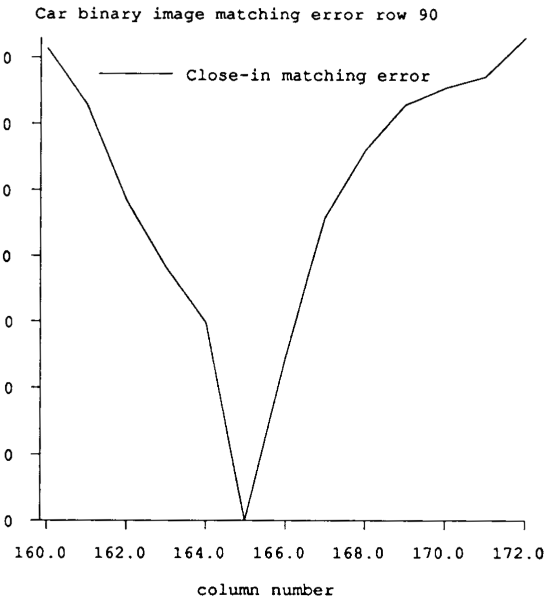

Fig. 5 Close-in Matching error

position, ie a fine search The column for the correct match was column 165. This point shows up very clearly on the graphs. The third figure shows the results for a coarse fine search where the template was moved either five columns or one column depending on the change in the error matching function. Again the correct column is clearly detected even when the partial search method is used. A magnified view of the search area is shown in the diagram fora full search, ie a calculation at all column positions.

Sparse Template Results.

As discussed, image search times may be reduced by speeding up the correlation calculation by a template image sub-sampling method. For this test, a full search was made for all sets of results. Results are as expected, with search times dropping rapidly with larger and larger sub-sampling grids. The sub sampling table shows search times for reduced row, column sampling as specified. It should be noted that the case taken was essentially ideal as no image noise was introduced to the image. With noiseless images, an extremely sparse template can be used but for practical applications, more sample points per template would be required.

Sparse Template Results

row, column spacing	search time seconds
1	1684
2	418
4	107
8	29.7
16	7.6

Results for binary image and template using a 4x4 sparse template.

Sparse template results were for an exhaustive column and row search, ie matching calculations were performed at all row and column positions.

Coarse/Fine Column Search Results

Column spacing	search time seconds
1	30.2
2	15.2
3	10.8
4	8.3
5	6.9
6	fail

Results table is for a binary image of a car and the car number plate as shown in Figures 1 and 2.

Coarse-fine Row Search Results

Row Spacing	search time seconds
1	30.2
2	16.6
3	12.9
4	9.6
5	7.6
6	6.8
7	7.0

Note that larger row steps will fail for very sparse grids. Searches were started at the top left of the image and stopped at row 90, column 165 where the template matching error was zero. For the coarse fine column search results, sparse template spacing was held at 8 for row and column spacing but at 4 for the coarse fine row search. All programs were written in C, running under Unix on a Sun 3 Workstation.

CONCLUSION

Two recognition speedup methods, known as a coarse fine search method and a sparse template technique have been proposed and tested for binary images . Tabulated results given in this paper show that these methods are workable for binary images and give at least a 16 times sparse template and a 4 to 1 coarse-fine column speed up respectively. The coarse fine row search was not reliable for coarser templates. It should be noted that these speedup factors depend on the type of image, natural scene, textured image, etc. Rosenfeld and Vandenberg have used a block average template method [11] but their method requires the computation of image block averages which this method does not. It should be noted that not all coarse-fine speedups and sparse template combinations will work together. In particular, very sparse grids and large coarse-fine steps may not work together. Also note that the speedup factors are for noise free images, and must be expected to be less for actual searches.While this example is applied to a translation only search, this work points the way towards speed up of scaling and orientation searches as well. This work was first applied to a number plate location search.

REFERENCES

[1] Caelli T.M. Liu Z.Q. "Multi object Pattern Recognition and Detection in Noisy Backgrounds Using a Hierarchical Approach" Computer Vision Graphics And Image Processing 44 pp296-306 1988

[2] Caelli T.M. Liu Z.Q "On the Minimum Number of Templates required for Shift, Rotation And Size Invariant Pattern Recognition" "Pattern Recognition" Vol 21, No 3 pp205-216 1988

[3] Cohen,H.A. Harvey,A.L "The Visual Estimation of Flow Rates of Aggregates and Slurries" IEEE International Conference in Image Processing ICIP-89 Singapore 5-8 Sept'89 pp811-814

[4] Grimson W.E.L. "Recognition of object families using parameterised models" Proc. Intnl Conf Computer Vision London UK, IEEE-CS 1987 pp93-101 1987.

[5] Gonzalez R. & Wintz P. "Digital Image Processing" Reading, MA Addison-Wesley 1977.

[6] Hall E. Computer "Image Processing & Recognition" New York NY Academic Press 1981.

[7] Harvey A.L. Cohen H.A. "Speed Up Methods for Image Object Recognition" ICARCV90 Singapore 1990 pp

[8] Harvey A.L. Cohen H.A. "Image Optical Flow Techniques for Speed Measurement." International Symposium on Signal Processing and Applications Queensland. 27-31 August 90 pp563-566.

[9] Horn B.K.P "Robot Vision" MIT Press. 1986

[10] Ikeuchi,K, Kanade,T."Automatic Generation of Object Recognition Programs" Proc.IEEE August pp1016-1035 1988

[11] Rosenfeld A. Vanderberg G.J "Coarse fine Template Matching IEEE Trans. Syst, Man, Cybern. pp104-107 1977

[12] Schalkoff,R "Digital Image Processing and Computer Vision" Publisher John Wiley New York Chapter5 pp210-253 1989