MVA'94 IAPR Workshop on Machine Vision Applications Dec. 13-15, 1994, Kawasaki

THE PARALLEL DETECTION OF INTERESTING POINTS IN DISTANCE TRANSFORM FOR IMAGE MATCHING

J. You[1,*] *E. Pissaloux*[2] *T. Dahlin*[1] *W.P. Zhu*[1]

[1]School of Computer & Information Science
University of South Australia, The Levels
South Australia, 5095
[2]Institut d'Electronique Fondamentale
Université Paris XI
Orsay Cedex 91405, France

ABSTRACT

This paper presents a parallel method to detect interesting points as image feature pixels in distance transform for image matching. Unlike the traditional methods in which edge detection is regarded as the pre-processing step to obtain a binary edge image for distance transform operation, in our approach the interesting points are detected as feature pixels by means of a dynamic thresholding procedure. The concept of remote procedure call(RPC) in distributed systems is introduced for the parallel implementation to achieve the speedup without specific software and hardware requirement. Thus a guided image matching system can be established by the hierarchical detection of interesting points in distance transform.

Key words and phrases: Image matching, distance transform, edge detection, interesting point, parallel implementation, remote procedure call(RPC).

1. INTRODUCTION

The matching techniques for object detection and recognition can be viewed as the measurements to determine the degree of resemblance between two objects that are superimposed on one another. Based on the level of image feature extraction, the matching algorithms developed in the past can be divided into three categories: pixel-based method, low-level feature based method and high-level feature based method. From the calculation simplicity and speed concerns, the pixel-based methods are more suitable in practice. In general, a match evaluation function is required to show the mapping between two images in terms of the certain feature description with the degree of similarity in their attributes. Usually the similarity of two descriptions is defined in the form of a cost function or a distance function, where these costs are expected to be minimized and are zero only if both descriptions are identical.

The conventional implementation of distance transform for image matching are based on the detection of edge pixels as image feature points. The well-known Chamfer matching was initially proposed by Barrow *et al* [1] aiming to develop a universally useful edge matching algorithm by finding the best fit of edge points from two different images based on distance transform and distance minimization. Such a technique was further extended by Borgefors[2] by introducing a hierarchical matching scheme. However, the processing is very time-consuming as more edge pixels are involved in distance transform for the purpose of accuracy. In order to reduce the number of feature points in image without losing information for distance transform, we propose that the detection of edge points can be replaced by the detection of interesting points associated with a dynamic thresholding selection procedure. Furthermore, the concept of remote procedure call (RPC) in distributed systems can be introduced for the parallel implementation to achieve the speedup without specific software and hardware requirement.

In this paper the concept of distance transform is introduced in Section 1, the detection of interesting points is detailed in Section 2, the application of interesting points in distance transform for matching measurement is proposed in Section 3 and the parallel implementation using remote procedure call(RPC) is detailed in Section 4. In the final section presents the test results and our conclusion.

2. THE CONCEPT OF DISTANCE TRANSFORM

The Distance Transform (DT) is such an operation which measures the distance of non-feature pixels to the nearest feature pixel while the feature pixels get the value zero. The purpose of the distance transform is to produce a numeric image whose pixels are labeled with the distance between each of them and their closest border pixel. At the early stage of this research, the distance between two pixels was defined by the length of the shortest 4-connected path such as city-block distance or 8-connected path such as max or chessboard distance between them. It is important that the distance transform used in matching algorithms should produce reasonable good approximation of the Euclidean distance as the matching measure is computed from the distance values. By investigating and comparing different methods for the implementation of Distance Transform, Borgefors[4] proposed to use iterated local operations for the high discriminating ability of the matching measure, where global distances in the image are approximated by propagating local distance,

*This work was partly supported by 1994 French Foreign Ministry International Postdoctoral Fellowship

i.e., distances between neighboring pixels, over the image. The propagation process can be implemented in either parallel or sequential. The sequential approach is referred to as "Chamfer" distance. In most cases a 3*3 pixel neighbourhood is considered and the two local distances in a 3*3 neighbourhood are the distance between horizontal/vertical neighbours and between diagonal neighbours. In our test 3-4 DT was applied, where the distances between horizontal/vertical neighbours and between diagonal neighbours within the 3*3 window are set to 3 and 4 respectively. Borgefors[4] claimed that the 3-4 DT provides better results than those with city block and 2-3 DT.

For a given binary edge image, the corresponding distance image is initialized by first setting each edge pixel as zero and each non-edge pixel as infinity. The sequential implementation of distance transform consists of two parts, first "forward transform" from left to right and from top to bottom; and then "backward transform" from right to left and from bottom to top. The following summarizes the given procedure, where D(i,j) represents the distance of pixel (i,j) in the binary image E of size M*N to the nearest edge pixel.

```
Initialization:

for i=1, ... , M do
for j=1, ... , N do
    if (i,j) is edge pixel then
       D(i,j) = 0
    else
       D(i,j) = infinity
                (large value available)

Forward Transform:

for i=2, .... M do
for j=2, ..., N do
    D(i,j) = minimum(D(i-1,j-1)+4, D(i-1,j)+3,
             D(i-1,j+1)+4, D(i,j-1)+3, D(i,j))

Backward Transform:

for i=M-1, ..., 1 do
for j=N-1, ..., 1 do
    D(i,j) = minimum(D(i,j), D(i,j+1)+3,
             D(i+1,j-1)+4, D(i+1,j)+3,
             D(i+1,j+1)+4)
```

It should be pointed out that calculation within local windows is applied in the above sequential distance transformation. Therefore such a process can be significantly speeded-up by parallel implementation.

3. THE DETECTION OF INTERESTING POINTS

It is noted that the distance transform used in matching requires a binary image for the operation. Traditionally edge points are considered as image feature points and used to create binary image. However, there exist several restrictions due to the misdetection, sensitivity to noise and redundency for efficient matching. In order to reduce the number of points during matching while still reserve the feature of the original image, we propose the use of interesting points in distance transform for matching. The detection of interesting points is based on the measure of how interesting a point is and an interesting point should be regarded as distinctness, invariance, stability, uniqueness and interpretability. Therefore, such points must be distinguishable from immediate neighbours and excludes points sitting on the same edge. In general, the detection of interesting points can be summarised as a three-step procedure:

- Selection of optimal windows. The selection is based on the average gradient magnitude within a window of prespecified size. Search for local maxima, while suppressing windows on edges and guaranteeing local distinctness. The measure used should also be invariant of rotation.
- Classification of the image function within the selected windows. The classification distinguishes between types of singular points such as corners, rings, spirals, and even isotropic texture based on a statistical test.
- Estimation of the optimal point within the window as the classification. The estimation is precise for corners and for the centers of circular symmetric features or spirals.

Moravec[5] suggested that a point is considered interesting if it has local maximum of minimal sums of directional variances. For a local window ranging from 4*4 to 8*8, the directional variances can be expressed as

$$I_1 = \sum_{i,j} (I(i,j) - I(i,j+1))^2$$
$$I_2 = \sum_{i,j} (I(i,j) - I(i+1,j))^2$$
$$I_3 = \sum_{i,j} (I(i,j) - I(i+1,j+1))^2$$
$$I_4 = \sum_{i,j} (I(i,j) - I(i+1,j-1))^2$$

where (i,j) represents the elements in the window. The interestingness of a point is then given by

$$I(i,j) = \min(I_1, I_2, I_3, I_4).$$

Thus a point whose local maximum is over a pre-set threshold will be considered good as an interesting point, where the pre-set threshold can be chosen based on the image histogram. In our test, the threshold is determined dynamically for optimal performance based on the interestingness histogram of the filtered image after Moravec operation.

4. THE PARALLEL DETECTION OF INTERESTING POINTS VIA RPC

The need for very high speed processing in practice of image processing means parallel solutions have to be explored. The parallelism means both functional parallelism and data parallelism. Obviously the execution time of a particular vision task can be reduced by parallel execution of multiple sub-tasks. On the other hand, for each computation intensive sub-task, the speed-up can be achieved by exploiting data parallelism. For example, the whole image can be divided into several subregions and processors are allocated to different regions for parallel operation.

The final processing result is then collected by combining the local computation results. The following lists the sub-tasks of our test:

- Step 1: The Detection of interesting points:
  The interesting points in the test image I are detected by using Moravec operator with local window size 4*4 which is outlined in the above section.
- Step 2: Distance transform:
  In this stage, a local mask 3-4 DT is applied to the interesting points detected image and the corresponding distance image is obtained.
- Step 3: The matching measurements:
  In our test the root mean square criteria is applied to the distance image for the matching measurement

It should be pointed out that both data parallelism and functional parallelism applicable to the detection of interesting points, where the calculation of I_1, I_2, I_3 and I_4 within a local window is performed simultaneously while the whole image is divided into sub-regions for the same operation. In our approach to the parallel implementation, remote procedure call(RPC) in distributed systems is introduced to share and transfer information on different machines for fast processing, which can be further applied to a heterogeneous system. In the initial test stage, the execution of Moravec operation for the detection of interesting points is performed by separating the implementation into two parts - servers and client. Figuer 1 shows the system structure for parallel implementation.

The operation for each server can be summarized as below:

• registers with its local name server

• sits and waits for RPC call

• performs Moravec operation when calls received and sends reply on completion

On the other hand, the client is the master which controls the parallelism in the following way:

• reads in original image data

• divides the image into sub-regions with some pixel overlap essential for data parallelism

• calls the four servers to perform Moravec operation

• receives replies from the servers

• combines the results to obtain the final image with detected interesting points

Our initial experimental result presented in Figure 2 shows that interesting points reduce the redundant image feature pixels for efficient matching by eliminating simple edge points which have no variance in the direction of the edge. Figure 2(a) is the original aerial image, Figure 2(b) is the edge detected image and Figure 2(c) is the interesting point detected image. The performance of both sequential and parallel detection of interesting points using Moravec operator is compared. On the average the sequential execution time for a 256*256 image implemented on Classic SPARC workstation is 6.2 sec. while the parallel implementation on a 4-process structure takes 5.0 sec.. The speed-up will be more effective when the algorithm is more complicated and more processes are involved for the implementation.

It should be emphasized that the RPC mechanism provides function-call semantics for local or remote interprocess communication which enables us to write application programs consisting of a set of procedures that do not all reside on a single computer but on different computers. Therefore, resources available to an application are no longer limited to a single computer and computing power can be added incrementally to the system. Our test data shows that the processing speed is increased by detecting interesting points to remove redundant edge pixels without any specific architecture requirement for parallelism. The speed is expected to be increased further if the load balancing requirements for the system are considered.

5. CONCLUSION

We conclude that the number of image feature pixels can be reduced without losing information for distance transform by detecting interesting points via an adaptive threshold selection procedure. The calculation burden can be further reduced by means of RPC for parallel implementation while no dedicated software and hardware architectures are required. When fully developed, the RPC mechanism for local or remote interprocess communication can be applied to various areas in image processing by sharing and transferring information on different machines for fast processing even in a heterogeneous system.

6. REFERENCES

[1] H.G. Barrow, J.M. Tenenbaum, R.C. Bolles and H.C. Wolf, "Parametric correspondence and chamfer matching: Two new techniques for image matching", *Proc. 5th Int. Joint Conf. Artificial Intelligence*, Cambridge, MA, pp. 659-663, 1977.

[2] G. Borgefors, "Hierarchical chamfer matching: a parametric edge matching algorithm", *IEEE Trans. Patt. Anal. Machine Intell.*, Vol. PAMI-10, pp. 849-865, 1988.

[3] D.P. Huttenlocher, G.A. Klanderman and W.J. Rucklidge, "Comparing images using the Hausdorff distance", *IEEE Trans. Patt. Anal. Machine Intell.*, Vol. PAMI-15, pp. 850-863, 1993.

[4] G. Borgefors, "Distance transforms in digital images", *IEEE Trans. Patt. Anal. Machine Intell.*, Vol. PAMI-8, pp. 334-343, 1986.

[5] D. Marr and E. Hildreth, "Theory of edge detection", *Proc. R. Soc. Lond. B*, Vol. 207, pp. 187-217, 1980.

[6] J. Shen and S. Castan, "An optimal linear operator for edge detection", *Proc. CVPR'86*, Miami, 1986.

[7] D.H. Ballard and C.M. Brown, *Computer Vision*, Englewood Cliffs, NJ: Prentice-Hall, 1982.

[8] A. Rosenfeld and A.C. Kak, *Digital Picture Processing*, Vol. 2, New York: Academic, 1982.

[9] J.R. Corbib, *The art of distributed applications* Springer-Verleg, 1991.

[10] J. You, E. Pissaloux, J.L. Hellec and P. Bonnin, "A guided image matching approach using Hausdorff distance with interesting points detection" to be appeared in the *Proc. of 1st IEEE international conference on image processing*, Austin, USA, November 13-16, 1994.

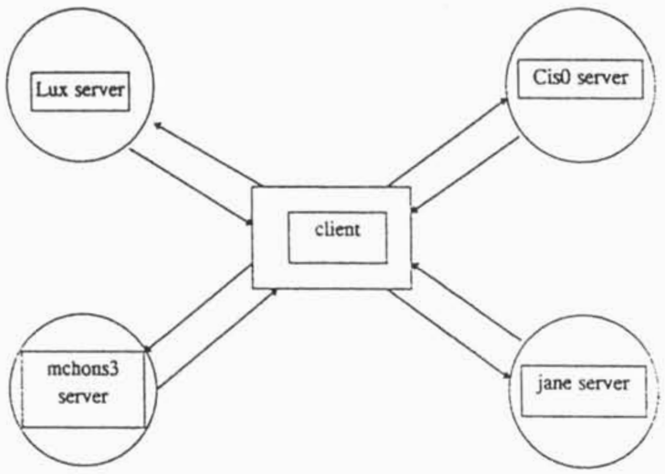
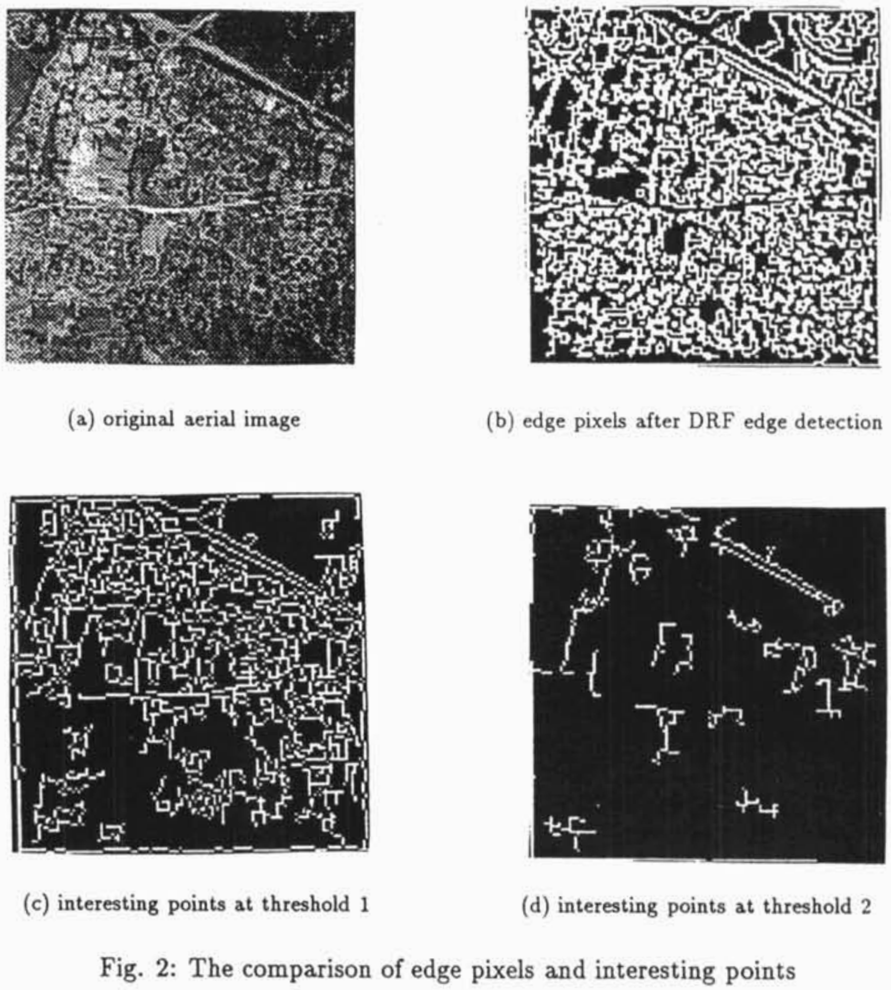


Fig. 1: The communication structure between client and servers during RPC calls



(a) original aerial image

(b) edge pixels after DRF edge detection

(c) interesting points at threshold 1

(d) interesting points at threshold 2

Fig. 2: The comparison of edge pixels and interesting points