A Parallel Matching Algorithm Based on Image Gray Scale

Liang Zong, Yanhui Wu
Faculty of Information Science and Engineering, Ningbo University, Ningbo, China
kevzong@hotmail.com

Abstract
In this paper, the idea of cluster parallel processing is taken in image gray scale matching. A data division processing algorithm based on MPI is applied to the known image gray scale matching. Experiment results show that parallel processing can shorten the overhead of gray scale matching significantly, and high speedup and efficiency can be acquired. This paper is expected to take technical insights into further investigation for parallel image processing based on cluster system.

1. Introduction
In the digital image processing, image matching is a process that according to the known image searches for the corresponding sub-image in a strange image. It is widely used in a variety of areas such as the computer vision, the medical images and the aircraft guidance. At present, there are a lot of image matching algorithms. The algorithm based on gray scale is of a simple and high precision, but it requires substantial computation and is sensitivity to the deformation [1]. The algorithm based on the characteristics needs not so much calculation, and owns better adaptability to the gray scale changes in the deformation and the shelter, but it depends on the quality of characteristics extraction, adding its matching precision is not very ideal [2]. The algorithm based on neural network and genetic has a good parallel, non-linear overall, a good memory and fault tolerance, but it requires substantial computation and its parameters of selected have a great impact on the result [3] [4]. Among them, the classic gray scale correlation algorithm features in high precision and easiness for hardware to achieve, but a great deal of calculation impose great restrictions on their application.

Nowadays, there are a lot of algorithms for improved gray scale matching, such as the gray normalized correlation matching. Basically they improved the algorithm on the function of similarity measurement [5], many of them are based on the serial processing. Along with the rapid development of computer hardware in recent years, the serial processing becomes possible to replace the traditional large scale workstation with the cluster that composed by multi-computer. Therefore the problem of substantial computation can be solved by the latter. In the study of image processing, the parallel processing based on the cluster greatly shortens overhead and becomes an important method. This paper based on the gray scale correlation matching takes parallel processing and improving.

2. Gray scale correlation matching
The known image is the template, given by T(m, n) and the size is N×N. The strange image given by S(m, n) and the size is M×M. The part of strange image covered by the template is the sub-image, given by S(m,n)_i and the size is N×N. In the strange image the start position of the sub-image is the (i, j). The matching course is that the sub-image searches for the known image in the strange image point by point. The common methods of gray scale correlation matching have the absolute difference, the mean absolute difference, the square difference and the mean square difference [3]. This paper that discusses the parallel processing uses the square difference as the basic algorithm, as follows:

\[
D(i, j) = \sum_{n=1}^{N} \sum_{m=1}^{N} [S(m,n) - T(m,n)]^2 \quad (1)
\]

where \( D(i, j) \) is the square difference, equation (1) expansion as follows:

\[
D(i, j) = \sum_{n=1}^{N} \sum_{m=1}^{N} [S^2(m,n) + T^2(m,n) - 2S(m,n)T(m,n)] \quad (2)
\]

From (1), we known that when the \( D(i, j) \) is smaller the accuracy between the \( T(m,n) \) and the \( S(m,n) \) is higher. In equation (2), the former two are the square sum of sub-image that the template corresponding to the strange image, as well as template. The \( S(m,n)_i T(m,n) \) is the mutual correlation of the template with sub-image. For the \( D(i, j) \) minimum, the \( S(m,n)_i T(m,n) \) must be maximum. The similarity measures of the sub-image and the templates are defined as:
\[ R(i, j) = \frac{\sum_{m=1}^{N} \sum_{n=1}^{N} [T(m,n) \times S(m,n)]}{\sum_{m=1}^{N} \sum_{n=1}^{N} S^2(m,n)} \] (3)

When \( R(i, j) \) is greater, the similarity is higher. After (3) standardization as follows:

\[ R(i, j) = \frac{\sum_{m=1}^{N} \sum_{n=1}^{N} [T(m,n) \times S(m,n)]}{\sqrt{\sum_{m=1}^{N} \sum_{n=1}^{N} T^2(m,n) \sqrt{\sum_{m=1}^{N} \sum_{n=1}^{N} S^2(m,n)}}} \] (4)

From above we know that when \( R(i, j) \) takes the maximum the \( D(i, j) \) will takes the minimal, therefore the most accurate location is \( (i, j) \).

3. Parallel model and implementation

In recent years, along with the rapid development of computer hardware and the performance improvement of network, the parallel computing can be transferred from the traditional supercomputer to the cluster system. The cluster system is a collection that consists of a group of independent computers (can be the personal computers), which are connected by the network. Each computer can be used as a single computer and also can be coordinated with other computer compos a centralized computing resource. The centralized computing resource can be used for the parallel task [6].

The parallel environment requires the support with software, hardware, and network. MPI is a standard of message passing interface for development bases on message passing parallel program, which aims to provide users with a practical, portable, efficient and flexible messaging interface library. MPI supports the C and FORTRAN programming language. It becomes the most popular standard of parallel development.

Under the parallel environment we must consider the parallel feasibility of the problem. In the process of solving the problem, the parallel characteristics must be fully reflected. The general steps of the parallel processing as follows:
1) Analyze the model of serial processing, find the largest part of calculation and analyse whether it can be parallel processing.
2) The master node divides the largest part of the calculation into the data blocks, sends the data blocks to the slave nodes. According to the relationship between the calculated results and the data blocks, decide to adopt which parallel processing mode.
3) Each slave node calculate, and sends the results to the master node.
4) The master node accepts the results of the slave calculate, compares to a single processing time and analyses whether the problem fits the parallel processing.

Because the image pixel is a two-dimensional array and the gray scale correlation matching deals with the pixels point by point, it could be considered as parallel processing. The algorithm can be described as follows:

1) The largest calculation of gray scale correlation matching is for the \( R(i, j) \). We can divide the strange image into \( p \) data blocks, each block has a continuous \( r \) row vectors, \( r = \left\lfloor \frac{M}{p} \right\rfloor \).
2) The master node sends \( p \) data blocks with “MPI_Send ()” to the \( p \) slave nodes that marked the 0, 1, ...(p-1). Each block calculates \( R(i, j) \) is independent and irrelevance with the other data blocks, so the algorithm takes the master-slave mode. The 0 node is the master node, which is responsible for sending \( p \) data blocks, accepting the results of the slave nodes calculate and calculating the results of the first block. As the calculation is searching for two-dimensional array fully, each node that include the 0 node to the (p-2) slave node needs to accept \( (r + N-1) \) row vectors.
3) The slave nodes marked 1, 2, 3 ... (p-1) accept data blocks which the master node sends with “MPI_Recv ()”. The nodes marked 1, 2, 3 ... (p-2) accept \( (r + N-1) \) row vectors and the final node (p-1) accepts \( M-(p-1) \times r \) row vectors. Each node calculates the \( R(i, j) \) and sends the results to the master node with “MPI_Send ()”.
4) The master node accepts the results with “MPI_Recv ()”, compares the \( R(i, j) \) and finds the most precise location of the gray scale matching.

4. Experiment results

Generally parallel computing performance takes the speedup \( S_p \) and the parallel efficiency \( f_p \) to measure, given by \( S_p = T_s / T_p \) and \( f_p = S_p / p \). The \( T_s \) is the single node running time (that is time of serial running). The \( T_p \) is the time of \( p \) nodes. The \( p \) is the number of node. This experiment takes Red Hat 9.0 for the operating system. The cluster is composed by 4 computers. The configurations as follows:
1) The master node CPU: Celeron 2.00GHz, Memory: 384M.
2) The slave nodes CPU: P4 1.5GHz, Memory: 256M.

The MPICH takes the MPICH- 1.2.7. By contrast, the size of the strange image adopts 256x256 pixels, 512x512 pixels and 1024x1024
pixels. The templates take the 32x32 pixels and 64x64 pixels.

Figure 1. Strange image and template image

(a) Strange image
(b) Template 32x32
(c) Template 64x64

Table 1. Overhead of template 32x32 (s)

<table>
<thead>
<tr>
<th>image size</th>
<th>node 1</th>
<th>node 2</th>
<th>node 3</th>
<th>node 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>256x256</td>
<td>1.53</td>
<td>0.83</td>
<td>0.59</td>
<td>0.42</td>
</tr>
<tr>
<td>512x512</td>
<td>6.81</td>
<td>4.71</td>
<td>2.53</td>
<td>1.91</td>
</tr>
<tr>
<td>1024x1024</td>
<td>32.51</td>
<td>18.03</td>
<td>12.01</td>
<td>9.34</td>
</tr>
</tbody>
</table>

Table 2. Overhead of template 64x64 (s)

<table>
<thead>
<tr>
<th>image size</th>
<th>node 1</th>
<th>node 2</th>
<th>node 3</th>
<th>node 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>256x256</td>
<td>4.31</td>
<td>2.74</td>
<td>1.62</td>
<td>1.23</td>
</tr>
<tr>
<td>512x512</td>
<td>23.86</td>
<td>13.42</td>
<td>8.64</td>
<td>7.18</td>
</tr>
<tr>
<td>1024x1024</td>
<td>116.29</td>
<td>72.59</td>
<td>48.63</td>
<td>34.73</td>
</tr>
</tbody>
</table>

5. Conclusions

In this paper, we propose an improved parallel algorithm for image gray scale matching. The algorithm is suitable for calculation intensive problems that usually spend much time on computation. Experiment results show that image gray scale matching is accurate. The algorithm can be used as a reference to image parallel processing. Our further works will focus on improving and optimizing the algorithm for better performance.

Acknowledgements

This work is supported by Science Foundation of Ningbo University (No.XK200583).

References