Image Object Tracking System

Using Parallel Mean Shift Algorithm

Sang Gu Lee
Department of Computer Engineering, Hannam University, Daejon, KOREA

Abstract - We implement a real-time image object tracking system with PTZ cameras. In general, mean shift algorithm is efficient for real-time tracking because of its fast and stable performance. However, in the image tracking system for PTZ cameras, the speed is not satisfied. So in this paper, we use parallel mean shift algorithm based on the color image distribution of detected object. In this system, MATLAB v.2012a and Parallel Computing Toolbox are used for CUDA computing in GPUs. This system can be applied to a faster image surveillance system for continuous object tracking in a wider area.

Keywords: PTZ camera, Object tracking, parallel mean shift algorithm, PCT, CUDA

2 (Parallel) mean shift algorithm

Mean shift algorithm is a general non-parametric mode clustering procedure. Mean shift image segmentation has 2 main steps as discontinuity preserving filtering and mean shift clustering [1]. Using Mean shift algorithm for real-time object tracking is reported in [2, 3]. The procedure of mean shift algorithm for a given point $x_i$ is as follows.

1. Computing the mean shift vector $mv(x_i)$
2. Translating density estimation window

$$x_i^{t+1} = x_i^t + mv(x_i^t)$$
3. Repeating step 1 and 2 until convergence

$$\forall f(x_i) = 0$$

Refer to [1, 2 and 3] about the details of the mean shift algorithm.

Recently, researches on the parallel mean shift algorithm are reported in [6, 7]. In [6], parallel dynamic mean shift algorithm based on path transmission is proposed. In this, input data set (S) and the mean data set (M) are updated by parallel calculations. In [7], by using K-means clustering to partition the object color space and lookup tables, object distribution with quite very small number of bins is obtained by parallel mean shift algorithm, and computations of the candidate histogram and calculation of the mean shift vectors are performed in GPUs. In this paper, we just use a parallel mean shift method based on vectorizing of the standard mean shift algorithm. We use CU file in C++ to be executed on the GPU and a compiled PTX file. With these 2 files, we create a kernel object in MATLAB as like parallel.gpu.CUDAkernel (‘para_ns.ptx’, ‘para_ns.cu’).

3 Proposed system

In the proposed system, images are captured from PTZ camera. These images are transferred to PC, and MATLAB...
program extracts moving object and eliminates the background images. Then, the preprocessing procedures such as filtering and morphological computations (erosion, dilation, open and close operations) are performed. MATLAB program send the packets such as P, T and Z data to the PTZ camera. In the monitor, the moving objects are displayed in the middle part of the screen. We use SPD-1000 PTZ Dome camera [4]. A frame size of 320 x 240 image data is used.

System development environment is as following as Table 1.

<table>
<thead>
<tr>
<th>Development environment in PC</th>
<th>Development environment</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hardware</strong></td>
<td><strong>Hardware</strong></td>
</tr>
<tr>
<td>CPU: Pentium (R) D 3.00GHz</td>
<td>CPU: Pentium (R) D 3.00GHz</td>
</tr>
<tr>
<td>Memory: 2GB</td>
<td>Memory: 2GB</td>
</tr>
<tr>
<td>Graphic card : GTX 560 Ti, 1GB</td>
<td>Graphic card : GTX 560 Ti, 1GB</td>
</tr>
<tr>
<td>PTZ camera : SPD-1000</td>
<td>PTZ camera : SPD-1000</td>
</tr>
<tr>
<td><strong>OS</strong></td>
<td><strong>OS</strong></td>
</tr>
<tr>
<td>Microsoft Windows XP</td>
<td>Microsoft Windows XP</td>
</tr>
<tr>
<td>Professional SP3</td>
<td>Professional SP3</td>
</tr>
<tr>
<td><strong>Software tool</strong></td>
<td><strong>Software tool</strong></td>
</tr>
<tr>
<td>MATLAB 2012a, PCT 6.0</td>
<td>MATLAB 2012a, PCT 6.0</td>
</tr>
<tr>
<td>CUDA development toolkit</td>
<td>CUDA development toolkit</td>
</tr>
</tbody>
</table>

![Fig. 1 GUI window for controlling PTZ](image1)

Fig. 1 shows a GUI window display to transfer packets for controlling motors of PTZ camera. Fig. 2 shows image clustering data of moving object after eliminating background and performing image pre-processing. In this figure, large cross marks display cluster's centroids. In this system, we only use the clusters in which total number of pixels is greater than 2500. We get the clustering data faster in parallelized method than that of non-parallelized method in 23 times faster.

4 Conclusions

In this paper, we implemented an image object tracking system for PTZ cameras using parallel mean shift method and CUDA computing. To detect moving object in the surveillance area, we used the parallel mean shift algorithm based on vectorizing technique of the standard mean shift algorithm. By using MATLAB language we controlled the PTZ camera and showed the trace of the object moving routes in real-time. Using the Nvidia GTX 560 GPU, a 23 speedup improvement (only part of mean shifting) can be made compared with original mean shift algorithm. For the future study, we will develop an entire parallel system including background computation and image pre-processing(open, close, and dilation operations) under CUDA environment. As a future research, we will develop a faster parallel mean shift algorithm with robust kernel function.

ACKNOWLEDGMENT

This work was supported by Security Engineering Research Center, granted by the Korea Ministry of Knowledge Economy.

5 References