Video Stabilization and Motion Deblurring on GPU

Kenji Takahashi[*]
Shizuoka University

Kenjiro T. Miura[†]
Shizuoka University

1 Introduction

As the video cameras used by amateur are relatively light and they are held by one hand, it is prone to be unstable. As a result the video sequences are affected by the vibrations of the video cameras and it is not comfortable to appreciate them.

We have achieved real time video stabilization by use of the graphical processing unit (GPU) which can perform parallel processing[Takahashi et al. 2008]. However we cannot remove motion blur recorded to each image of the video sequence caused by the movement of the video camera and the processed video includes unnatural blur without vibration. We will propose a new method for motion deblurring in this paper

2 Procedure

At first, we detect global motion of the video camera and stabilize video based on the detection. We assume that the movement of the camera after removal of the undesirable vibration is smooth. For the detected motion, we remove vibration by the Gauss function. To interpolate undefined regions produced by the correction of global motion in which the pixels are outside the recorded image, we perform mosaicking. We have implemented two methods; inverse convolution and unsharpening mask which are selectable for certain purposes to motion deblurring caused by the camera vibration.

2.1 Implementation on GPU

The video sequence is input from a file or a camera. The GPU processes data transfered from the CPU. However the GPU cannot input directly from a hard disk drive or a camera. Hence at first the video is input to the main memory of the CPU and it is transfered to the memory (VRAM) of the GPU. Then the GPU can perform parallel processing. We use CUDA (Computer Unified Device Architecture) to implement programs on the GPU to use its functions easily.

2.2 Global motion estimation

It is necessary to detect global motion of the camera to stabilize the video. By comparing two consecutive images of the video sequence, the camera motion can be detected[Takahashi et al. 2008].

2.3 Motion deburring

Motion blur cannot be removed by the method mentioned so far. We will remove motion blur caused by the exposure time of the camera and the sensitivity of the CCD.

2.3.1 Inverse convolution

We assume that in the Fourier space $F(u)$ is an image without motion blur, $H(u)$ is a PSF (Point Spread Function) and $G(u)$ is an image with motion blur. The image without motion blur is obtained by the following equation.

$$\frac{G(u)}{H(u)} = \frac{1}{H(u)}F(u)H(u) = F(u) \qquad (1)$$

Therefore if the PSF is known, it is possible to remove motion blur from the image. However the PSF for the video sequence is unknown, we have to estimate it accurately. We have developed a new method to estimate the PSF accurately.

2.3.2 Unsharpening Mask

The unsharpening mask is a sharpening filter constructed by subtracting unsharpened (smoothed) image from the original image. In this research we make it directional as shown in Fig.1 corresponding to the motion blur direction We not only deal with 9 pixels bu also dynamically adjust the size of the pixels for the filter

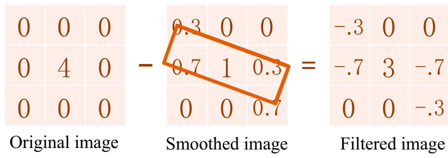

Figure 1: *Unsharpening mask for 9 pixels*

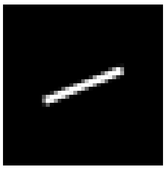

Figure 2: *Kernel*

3 Results

Table 1 shows processing times for video stabilization of different sizes. We used the BGFS method for minimization search for all videos. We have achieved 300 times speed up by use of the paralell calculation power of the GPU.

Table 1: *Processing times for video stabilization of different sizes*

Size[pixel]	CPU[fps]	GPU[fps]	GPU/CPU
512×512	0.2006	46.49	231.8
768×768	0.08806	25.03	284.2
1024×1024	0.05025	14.68	292.1

4 Consclusion

In this paper we have proposed a method to stabilize and motion deblur of high-resolution video images. We have achieved 32 fps for 640 × 480-pixel (VGA) video processing including all of global motion estimation on the GPU, mosaicking and motion deblurring by the unsharpening mask.

References

LITVIN, A., KONRAD, J., AND KARL, W. C. 2003. Probabilistic video stabilization using kalman filtering and mosaicking. In *IS&T/SPIE Symposium on Electronic Imaging, Image and Video Communications*, 663–674.

TAKAHASHI, K., FUJISAWA, M., AND MIURA, K. 2008. Real time video stabilization on gpu. In *Proc. Visual Computing/Graphics and CAD Joint Symposium*.

[*] e-mail:f0830037@ipc.shizuoka.ac.jp
[†] e-mail:tmkmiur@ipc.shizuoka.ac.jp