2018 IEEE International Conference on Consumer Electronics (ICCE)

GPU-Accelerated High-Resolution Image Stitching with Better Initial Guess

Chia-Ho Lin⋆1, Yuhsiang M. Tsai⋆2, Weichung Wang2, Liang-Gee Chen1, *Fellow*, IEEE
1DSP/IC Design Lab, Institute of Electronics Engineering
2Institute of Applied Mathematical Sciences
National Taiwan University, Taipei, Taiwan
Email: {chiahoo2004, lgchen}@video.ee.ntu.edu.tw, {yhmtsai, wwang}@ntu.edu.tw

***Abstract*—High-resolution image processing is crucial for emerging applications in virtual reality. The trade-off between robustness/accuracy and speed pose a challenge to high-resolution image stitching. This paper accelerates the state-of-the-art image stitching algorithm by using the CUDA toolkit. We also use the mesh coordinates as an initial guess for solving warping function for an additional increase in speed. Experimental results show that our implementation achieves comparable quality to the state-of-the-art work, while our implementation is almost 2x faster on high-resolution images (3264x2448).**

***Index Terms*—image stitching, panorama, image warping, acceleration, CUDA**

I. Introduction

Virtual reality (VR) can be widely used for many applications including video gaming, simulated training and assistance for medical, combat and vehicle operation. Best-selling headsets such as the Oculus Rift generally deliver 2K resolution, which is not enough for many VR applications. Users can still see aliasing on lines on a retina display, which is bad for user experience [1]. Recent years, researches have focused on improving the resolution of VR headsets. The world's first 8K VR headset was released this year [2]. It is believed that more devices for 8K or even 16K resolution will be developed [1], which makes high-resolution image processing crucial in the future.

Panoramic image stitching used to create virtual environment is the necessary technology of image processing for VR applications. The trade-off between quality and speed is an important issue for image stitching algorithm. The ability to deal with the big difference in shot angles is the key to produce good panorama. Recently, several algorithms [3]–[6] sacrifice the speed for better quality and robustness. [6] combines the advantages of these methods and achieves state-of-the-art quality. However, the large computational cost makes it slow for high-resolution images.

In this work, several techniques are used to accelerate the state-of-the-art algorithm [6], including GPU-accelerated parallel implementation for feature matching and better initial guess for solving warping function. Experimental results show that our implementation is almost 2x faster than the original C++ implementation on high-resolution images (3264x2448), but still preserves original quality.

⋆ indicates equal contribution

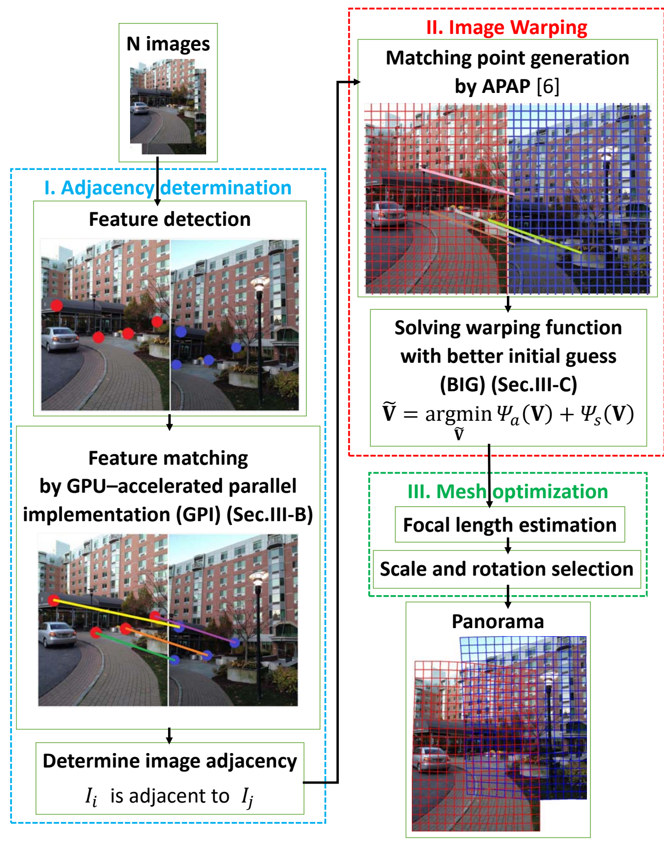

Fig. 1: The block diagram of our system.

The rest of the paper is organized as follows: Sec.II lists the related works. Sec.III introduces the proposed method. Sec.IV shows our results and we conclude in Sec.V.

II. Related Work

Although image stitching has been studied over ten years, how to produce good images from stitching still remains an open problem. Well-known methods such as AutoStitch [7] assume that the images have the same center of projection, which is rarely satisfied for VR panorama rendering. Several works have proposed solutions to this problem in recent years. Zaragoza et al. [3] proposes as-projective-as-possible warps which are globally projective, yet allow local non-projective deviations for better images alignment. Chen et al.

Computation	[6]	Ours
Feature matching	38%	3%
Solving warping function	25%	18%
Scale and rotation selection	19%	47%
others	18%	32%

TABLE I: The profiling result.

Resolution	Iteration number		Run time improvement w.r.t. [6]		
	[6]	BIG	GPI	BIG	GPI+BIG
1532x1022	342	200	23.3%	12.6%	31.6%
2000x1329	410	250	26.2%	14.2%	35.2%
3264x2448	568	300	33.3%	17.1%	41.7%

TABLE II: The comparison results of iteration number and run time improvement between proposed acceleration techniques and [6] in different resolutions.

[6] uses global similarity prior to prevent unnatural rotation. However, large computational cost of these methods increases the run time when image resolution is large. Since the quality and speed of image stitching are both important for VR applications, we propose an accelerated version of [6], which achieves state-of-the-art quality.

III. Approach

The block diagram of our system is shown in Fig. 1. The first phase, adjacency determination, finds the image pair that is possible to be adjacent in the final panorama. Given a set of N images, $I_1, I_2, ...I_N$, the feature detection module produces descriptors for each image by SIFT. The feature matching module then calculates the distance between descriptors. If the distance is smaller than a threshold, two descriptors are recognized as a matching pair. The system determines the image pair that is most likely to be adjacent in panorama based on the number of matching pairs.

The second phase is mesh-based image warping. We apply as-projective-as-possible warp (APAP) [3] to produce the mesh-based warping function. We use the initial mesh coordinates $\mathbf{V}$ and set of matching pairs obtained from the previous phase to guide the mesh deformation. The best mesh coordinates $\widetilde{\mathbf{V}}$ for stitching is determined by solving energy function $\widetilde{\mathbf{V}} = \underset{\widetilde{\mathbf{V}}}{argmin}\ \Psi_a(\mathbf{V}) + \Psi_s(\mathbf{V})$, where $\Psi_a(\mathbf{V})$ and $\Psi_s(\mathbf{V})$ represent the alignment term and the similarity term respectively.

The final phase determines the best scale and rotation for each image. Our system first estimates the intrinsic matrix, which contains the information of focal length, camera pixel size and skew parameters. Then we get the 3D rotation matrix and corresponding rotation angle for each image by solving equation with SVD method, and use it to optimize the mesh coordinates. Finally, the image is warped using rotation matrix and stitched to produce final panorama.

A. Profiling

To identify which part of the system is time-consuming, we use profiler to analyze it. The profiling result of [6] is shown in Table. I. Feature matching, solving warping function, scale and rotation selection consume most of the run time, which occupy 38%, 25%, 19% of run time respectively. So we focus on improving these parts for acceleration. Techniques to accelerate these performance bottlenecks are described in the following subsections.

B. GPU-accelerated Parallel Implementation (GPI)

To match the feature descriptors between images, we have to calculate the distance between each pair of descriptors. When the dimension of descriptor is high (usually larger than 500), memory access becomes time-consuming. GPU can efficiently access device memory when performing arithmetic operations on vectors or matrices. To fully exploit the parallelism in CUDA, we transform the distance computation to basic vector and matrix operations.

First, we store descriptors of image I_i and I_j in matrices $A = [\mathbf{x_1}, \mathbf{x_2}, ..., \mathbf{x_p}]$ and $B = [\mathbf{y_1}, \mathbf{y_2}, ..., \mathbf{y_q}]$ respectively, where p, q is the number of feature points detected in image I_i and I_j. The first column of A is $\mathbf{x_1}$, i.e. $A(k,1) = \mathbf{x_1}(k),\ k \in N,\ 1 \le k \le K$, where K is the dimension of feature descriptors. Rewrite the math expression of d using linear algebra:

$$\begin{aligned} d^2 &= \|\mathbf{x_p} - \mathbf{y_q}\|_2^2 \\ &= \sum_{k=1}^{K} (A(k,p) - B(k,q))^2 \\ &= \sum_{k=1}^{K} A(k,p)^2 - 2M(p,q) + \sum_{k=1}^{K} B(k,q)^2 \\ &= \|\mathbf{x_p}\|_2^2 - 2M(p,q) + \|\mathbf{y_q}\|_2^2,\ where\ M = A^T B \end{aligned} \tag{1}$$

Basic Linear Algebra Subprograms (BLAS) are routines that perform basic vector and matrix operations such as Euclidean norm and matrix multiplication, and the cuBLAS library is a fast GPU-accelerated implementation of BLAS. In the final expression, $\|\mathbf{x_p}\|_2$ and $\|\mathbf{y_q}\|_2$ can be calculated in parallel by using cuBLAS level 2 routines, and $A^T B$ by cuBLAS level 3 routines.

C. Better Initial Guess (BIG)

After matching the feature descriptors between images, Chen et al. [6] uses mesh-based image warping to stitch image. The mesh coordinates $\mathbf{V}$ is determined by selecting grid points along x and y axis with an arbitrary interval. In our case, the interval is 40 pixels. We collect I_i's mesh coordinates in the overlap region of adjacent images I_i, I_j as the set of matching points, $\mathbf{M_{ij}}$. For each matching point in $\mathbf{M_{ij}}$, the correspondence in I_j is identified by APAP [3]. $\mathbf{V}$ and $\mathbf{M_{ij}}$ are then used to construct the alignment term $\Psi_a(\mathbf{V})$ and the similarity term $\Psi_s(\mathbf{V})$. The goal is to find the best mesh coordinates $\widetilde{\mathbf{V}}$ that minimize the energy function. Chen et al. [6] solves this linear system by the conjugate gradient (CG) method, which is an iterative method. Generally, the iteration start with x=0. Beginning with better initial guess makes the convergence faster. In our observation, $\mathbf{V}$ (mesh

Fig. 2: The intermediate result after running 150 iterations. The left image is the result of proposed better initial guess, the mesh coordinates. The right image is the result of [6], which starts with x=0 as an initial guess.

coordiates *before* warping) is close to $\widetilde{\mathbf{V}}$ (mesh coordinates *after* warping). To speed up the convergence, we use $\mathbf{V}$ as initial guess. As shown in Fig. 2, our implementation converges faster when the number of iterations is the same.

IV. Experiment

We use the testing images provided by Chen et al. [6]. All results are performed on a 2.3 GHz Intel Xeon E5-2670 v3 CPU with 252GB of memory and NVIDIA Tesla K40C. To evaluate the performance improvement for different resolutions, we produce panorama of 3 images in three different resolutions: 1532x1022, 2000x1329, 3264x2448. The experiment results compared to [6] in different resolutions are shown in Table. II. With the help of better initial guess (BIG), we can reduce 50% of iteration numbers with respect to the original setting in [6] but still preserve original quality. For example, when the input is 3 images of 1532x1022, solving with BIG produces panorama with almost the same quality after 200 iterations, while [6] needs 342 iterations. The original implementation takes 21.7s, 33.4s, 83.5s for stitching 3 images of 1532x1022, 2000x1329, 3264x2448. After applying all proposed acceleration techniques including GPU-accelerated parallel implementation (GPI) and better initial guess (BIG), we achieve 31.6%, 35.2%, 41.7% run time improvement in different resolutions respectively. It is noticed that the larger input image size is, the more effective our proposed acceleration techniques are. The final profiling is shown in Table. I. As we can see, the running time spending on feature matching and solving warping function are reduced by 30% and 7%. Besides acceleration techniques including GPI and BIG, we also remove duplicate computation when solving the linear system. This makes run time spending on many other part lower. Scale and rotation selection becomes the main bottleneck.

V. Conclusion

It is believed that robust and fast image stitching method for high-resolution images is the key to emerging applications in VR. In this paper, we propose GPU-accelerated parallel implementation for feature matching and better initial guess for solving warping function to speed up the state-of-the-art method for image stitching. Experimental results show that our implementation is almost 2x faster than the original implementation while preserving original panorama quality on high-resolution images (3264x2448).

References

[1] "improving resolution." [Online]. Available: https://arstechnica.com/gaming/2013/09/virtual-perfection-why-8k-resolution-per-eye-isnt-enough-for-perfect-vr/

[2] "Pimax 8k vr." [Online]. Available: http://www.pimaxvr.com/

[3] J. Zaragoza, T.-J. Chin, M. S. Brown, and D. Suter, "As-projective-as-possible image stitching with moving dlt," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2013.

[4] C.-H. Chang, Y. Sato, and Y.-Y. Chuang, "Shape-preserving half-projective warps for image stitching," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[5] C.-C. Lin, S. U. Pankanti, K. Natesan Ramamurthy, and A. Y. Aravkin, "Adaptive as-natural-as-possible image stitching," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[6] Y.-S. Chen and Y.-Y. Chuang, "Natural image stitching with the global similarity prior," in *Proceedings of European Conference on Computer Vision (ECCV 2016)*, October 2016, pp. V186–201.

[7] M. Brown and D. Lowe, "Recognising panoramas," in *Proceedings of the 9th International Conference on Computer Vision*, vol. 2, Nice, October 2003, pp. 1218–1225.