











In the parallel implementation on GPU using CUDA, we configure the thread hierarchy as 64 thread blocks with 64 threads in each thread block. The average retrieval time for the two features on GPU is shown in Fig. 8 and Fig. 9 respectively. As we can see, the simple parallel implementation obtains 7-8 times speedup. Furthermore, the optimized implementation with load balancing accelerates the speed almost 40% further, though the parallel scan step for load balancing increases the overhead. The optimized implementation can retrieve in tens of milliseconds on modern GPU, meeting the need of real-time applications.

We note the differences in hardware implementations of the computer units between CPU and GPU. For example, the GPU device in our experiments only supports single precision floating-point numbers, while the CPU supports double precision floating-point numbers. Therefore, we check the effect on the results of retrieval. Fortunately, all the retrieval results under both the CPU and the GPU are always coincident in our experiments. These results demonstrate that single precision is enough in checking the spatial configurations of features.

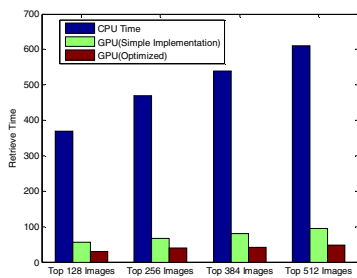


Fig. 8. Speedup on GPU(the MSER features).

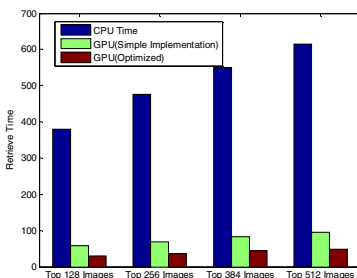


Fig. 9. Speedup on GPU(the Harris-Affine features).

## 5.CONCLUSION AND FUTURE WORK

In this paper, we propose a novel spatial matching method ACN-RANSAC for visual words based object retrieval and successfully map it to modern GPU with CUDA in parallel. ACN-RANSAC outperforms previous methods: Firstly, it is more robust than previous methods as it abandons the gravity vector assumption made in previous methods. Secondly, it decreases the computational cost, as it eliminates possible false matches beforehand and does not perform least square solutions in estimating 6 dof affine

transformations. In addition, the parallel implementation on modern GPU obtains ten times speedup. In the future work, we consider using multi-GPU devices to accelerate the retrieval speed further.

## REFERENCE

- [1] J. Sivic, A. Zisserman. Efficient visual search of videos cast as text retrieval. PAMI, 2009, 31(4): 591 - 606.
- [2] J. Sivic, A. Zisserman. Video google: A text retrieval approach to object matching in videos; ICCV, Washington D C, 2003.
- [3] D. Nister, H. Stewenius. Scalable recognition with a vocabulary tree; CVPR, New York 2006.
- [4] O. Chum, M. Perdoch, J. Matas. Geometric min-hashing-finding a (thick) needle in a haystack; CVPR, Miami, 2009.
- [5] H. Jegou, M. Douze, C. Schmid. Hamming embedding and weak geometric consistency for large scale image search; ECCV, Marseille, 2008.
- [6] J. Philbin, O. Chum, M. Isard, et al. Object retrieval with large vocabularies and fast spatial matching; CVPR, Minneapolis, 2007.
- [7] O. Chum, J. Matas. Locally optimized ransac DAGM symposium Magdeburg. 2003: 236-243.
- [8] D.G. Lowe. Object recognition from local scale-invariant features; ICCV, Kerkyra, 1999.
- [9] M. Perdoch, O. Chum, J. Matas. Efficient representation of local geometry for large scale object retrieval; CVPR, Miami, 2009.
- [10] M.A. Fischler, R.C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Communications of ACM, 1981, 24(6): 381-395.
- [11] Nvidia. Nvidia cuda programming guide version 2.2. 2008.
- [12] J.D. Owens, D. Luebke, N. Govindaraju, et al. A survey of general-purpose computation on graphics hardware. Computer Graphics Forum, 2007, 26(1): 80 - 113.
- [13] F. James, M. Steve. Computer vision signal processing on graphics processing units; ICASSP Montreal PQ 2004.
- [14] C. Terboven, T. Deselaers, C. Bischof, et al. Shared-memory parallelization for content-based image retrieval; ECCV, 2006.
- [15] K. Mikolajczyk, T. Tuytelaars, C. Schmid, et al. A comparison of affine region detectors. IJCV, 2006, 65(1): 43-72.
- [16] K. Mikolajczyk, C. Schmid. Scale & affine invariant interest point detectors. IJCV, 2004, 60(1): 63-86
- [17] D.G. Lowe. Distinctive image features from scale-invariant keypoints. IJCV, 2004, 60(2): 91-110.
- [18] M. Harris. Optimizing parallel reduction in cuda. 2008: NVIDIA Developer Technology.
- [19] M. Harris, S. Sengupta, J.D. Owens. Parallel prefix sum (scan) with cuda. GPU Gems 3. Addison Wesley. 2007: 851-876.
- [20] W. Fang, K.K. Lau, M. Lu, et al., Parallel data mining on graphics processors: The Hong Kong University of Science and Technology, 2008.