Evaluations on Processing Time

Josef Maier¹, Martin Humenberger¹, Markus Murschitz¹, Oliver Zendel¹, and Markus Vincze²

¹ AIT Austrian Institute of Technology
{josef.maier.fl, martin.humenberger, markus.murschitz.fl, oliver.zendel}@ait.ac.at
² Vienna University of Technology
vincze@acin.tuwien.ac.at

Abstract. In this document, we show additional results on processing time for 8 different matching algorithms. The evaluations are performed on various datasets for SIFT features as well as FAST keypoints & FREAK descriptors.

Fig. 1. Runtime analysis on the “wall” dataset from Mikolajczyk et al. [1, 2] using (a) FAST keypoints & FREAK descriptors and (b) SIFT features. For comparison, the following algorithms were used: Guided Matching based on Statistical Optical Flow (GM), hierarchical clustering tree (HC), linear matching (LI) from the FLANN library, Locality Sensitive Hashing (LSH), SparseVFC (VF) in combination with the hierarchical clustering tree, CasHash (CH), priority search k-means tree (HK), and the randomized KD-tree (RA). Time measurements were performed using the smallest runtime of 100 runs on an Intel Xeon E5-2687W 3.1GHz CPU. Each datapoint stems from a different image pair with an inlier ratio of 75%. The spikes of the runtime marked by a dashed black circle originate from switching to the fall-back similarity-based matching instead of guided matching, as the inlier ratio tendency factor $\varphi_e$ was below its threshold (0.08 for binary and 0.2 for high dimensional features). The fall-back solution is not triggered by the low number of features but by the difficulty of the evaluated scene.
Fig. 2. Runtime analysis on the KITTI flow dataset from Menze and Geiger[3] using (a) FAST keypoints & FREAK descriptors and (b) SIFT features. For comparison, the following algorithms were used: Guided Matching based on Statistical Optical Flow (GM), hierarchical clustering tree (HC), linear matching (LI) from the FLANN library, Locality Sensitive Hashing (LSH), SparseVFC (VF) in combination with the hierarchical clustering tree, CasHash (CH), priority search k-means tree (HK), and the randomized KD-tree (RA). Time measurements were performed using the smallest runtime of 100 runs on an Intel Xeon E5-2687W 3.1GHz CPU. Each datapoint stems from a different image pair with an inlier ratio of 75%.

Fig. 3. Runtime analysis on the KITTI disparity dataset from Menze and Geiger[3] using (a) FAST keypoints & FREAK descriptors and (b) SIFT features. For comparison, the following algorithms were used: Guided Matching based on Statistical Optical Flow (GM), hierarchical clustering tree (HC), linear matching (LI) from the FLANN library, Locality Sensitive Hashing (LSH), SparseVFC (VF) in combination with the hierarchical clustering tree, CasHash (CH), priority search k-means tree (HK), and the randomized KD-tree (RA). Time measurements were performed using the smallest runtime of 100 runs on an Intel Xeon E5-2687W 3.1GHz CPU. Each datapoint stems from a different image pair with an inlier ratio of 75%.
References