Fast Visual Hull Extraction at High Resolution

H. Scharr [1,*], C. Briese [1], P. Emgenbroich [1], A. Fischbach [1], F. Fiorani [1], M. Müller-Linow,[1]
Institute of Bio- and Geosciences (IBG), IBG-2: Plant Sciences, Forschungszentrum Jülich, Germany
h.scharr@fz-juelich.de

Measuring plant geometry from single view-point 2D images often suffers from insufficient information, especially when plant parts occlude each other (self-occlusion). In order to receive more detailed information and recover the plants 3D geometric structure volume carving is a well established method to generate 3D point clouds of plant shoots [6, 3, 2, 7], seeds [12, 13, 4], and roots [1, 17, 14]. Volume carving can be applied in high-throughput scenarios [3]: For the reconstruction of relatively simple plant structures like tomato seedlings image reconstruction takes approx. 25-60ms, based on a well though out camera geometry using 10 cameras and a suitably low voxel resolution $240 \times 240 \times 300$ voxels at 0.25 mm/voxel. Short reconstruction times are achieved by precomputing voxel to pixel projections for each of the fully calibrated cameras. However, precomputing lookup-tables is not feasible for high voxel resolutions due to storage restrictions [8]. Current implementations popular in plant sciences suffer from high computational complexity, when voxel resolutions are high. We therefore implemented and tested a fast and reliable volume carving algorithm based on octrees (cmp. [6]) and integral images (cmp. [15]). This work summarizes and extends our findings presented in [2].

Visual hull reconstruction via volume carving is a well-known shape-from-silhouette technique [10, 11, 9] and found many applications. Also octree as multigrid approach and integral image for reliable and fast forground testing have been used successfully with volume carving in medical applications [8] and human pose reconstruction [5]. Realtime applications at 512^3 voxel resolution have been achieved where suitable caching strategies on GPUs can be applied e.g. for video conferencing [16]. Here we show how even higher spatial resolutions are achievable on consumer computer hardware without prohibitively large computational or storage cost. Subsequent octree-voxel-based processing allows extraction of plant structural features suitable for phenotyping purposes.

Our main finding is that clearly the combination of octrees and integral images speeds up processing especially at very high resolutions compared with non-adaptive brute force implementations. However, multiresolution can only be fully effective, when applied using a suitable refinement strategy. Neither naïve breadth first nor depth first refinement strategies lead to significant speedup. Looping all images and marking octree candidate nodes for refinement, followed by a refinement step only when all candidates are marked is the key for speedup. We observed a speed up of up to a factor of 35 between these strategies, where CPU parallelization (8 cores) yields another factor of 3.

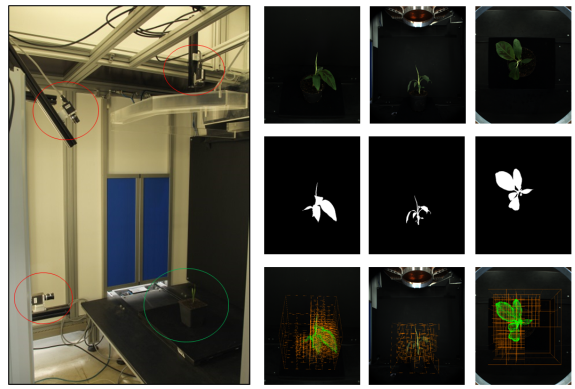

Figure 1. Imaging. Left: Camera setup in measurement chamber. Three 5MP RGB cameras (red circles) with different view angles and rotating table (green circle). Right, top: Original RGB images taken from 3 different view angles; middle: Binary masked images; bottom: Intermediate carving step overlaid on images.

In Figure 1 the imaging setup for our experiments is described. The imaged working volume is about $(50\text{cm})^3$ and we chose this as reconstruction volume. In contrast to reconstruction methods using uniform grids, using larger reconstruction sizes with multiresolution grids like octrees, does not result in (significantly) higher computation times or storage needs.

In Figure 2 a result is shown for a Banana seedling of 30cm height and width, reconstructed at finest voxel size of 0.12mm. A uniform voxel grid of this resolution would have $4096 \times 4096 \times 4096$, i.e. $6.9 \cdot 10^{10}$ voxel and could not even be stored in the available 8GB RAM of the computer used here, let alone carved in reasonable time. For comparison: In [13] a runtime of 12.5s is reported using a well optimized GPU implementation for reconstruction of