A Toolchain for Computer Vision Algorithm Development

Graduate Candidate

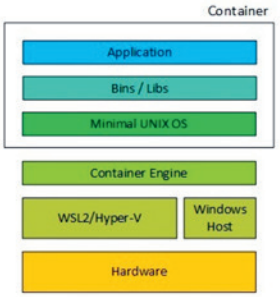
Andreas Kuhn

Problem: After initial prototyping of computer vision algorithms in Python, it often makes sense to implement the time-critical section of the algorithm in C/C++, to be able to assess the full performance potential. For development of computer vision applications in general and especially for a mixed language workflow as described above, many different tools and technologies are involved. Building, configuring and integrating these technologies into a cohesive development environment for the project-specific needs are elaborate, time-consuming and system-dependent task which have made exploring this area of computer vision development burdensome in practical labs if the computer software is being updated or changed.

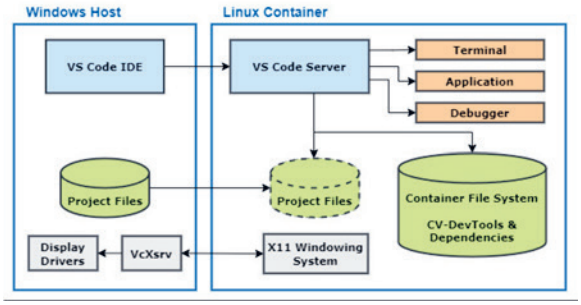
Approach: The Docker containerization technology is used to package the complex toolchain and the practicality of such a solution is evaluated. To streamline development on the container, options for accessing and interacting with the container are evaluated and integrated. The development environment is extended beyond Docker by including a CI/CD pipeline on GitHub, where the integrity of the container as well as the application developed within it are tested.

Result: Docker was proven to be a valid solution for the posed problem as well as for streamlining computer vision software development in general. The final solution provides a local-quality development environment including access to local hardware such as the GPU, while simultaneously being highly portable.
Two container images were built: The first is a lightweight solution providing the base toolset for developing mixed language projects using OpenCV, CMake and Pybind11. The second image additionally integrates the CUDA Toolkit as well as the OpenCV CUDA modules for building custom CUDA kernels to be executed on a Nvidia graphics card.
For each of the containers, a sample application was developed, showcasing usage as well as the efficacy of C++ and CUDA acceleration.

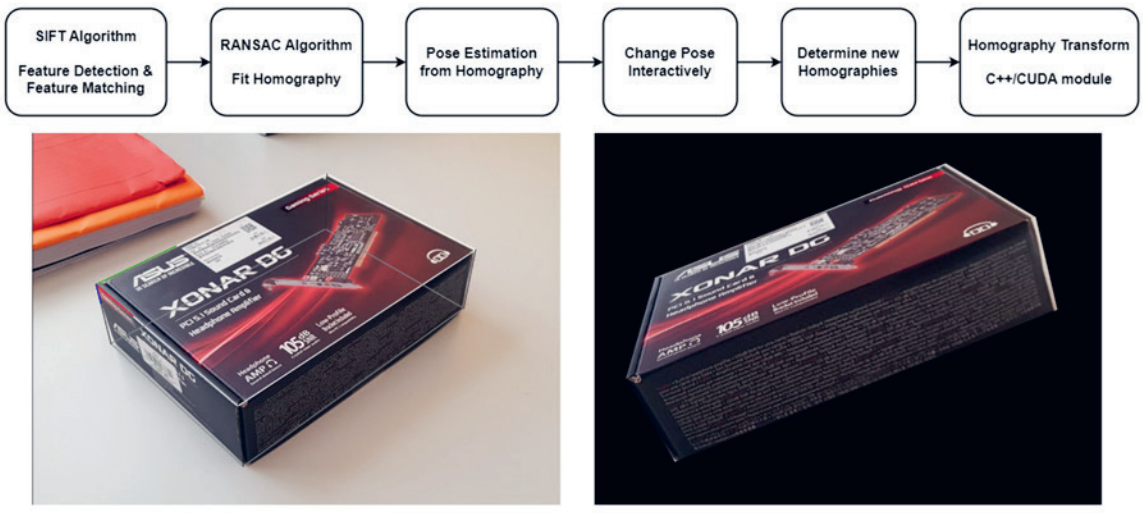
Docker container technology stack.
Own presentment



CV-DevContainer Toolchain.
Own presentment



Mixed Python/C++/CUDA application: Input image with wireframe; interactive real-time (5 ms) re-render in arbitrary pose.
Own presentment



Advisor
Prof. Dr. Martin Weisenhorn

Co-Examiner
Jonas Schmid, Stettbacher Signal Processing, Dübendorf, ZH

Subject Area
Image Processing and Computer Vision