

Developing embedded systems faster

September 2 2019, by Ulrich Pontes



Drone with stereo camera. The small white box holds the embedded system, which evaluates the slightly offset images from the two cameras in real time in order to detect obstacles. Credit: Fraunhofer IOSB

Whether for the car or the drone: Developing image processing software for embedded systems takes a lot of time and is therefore very expensive. Now the Tulipp platform makes it possible to develop energy-

efficient embedded image processing systems more quickly and less expensively, with a drastic reduction in time-to-market. The Fraunhofer Institute for Optronics, System Technologies and Image Exploitation IOSB is a member of the EU consortium which simplified the process.

At first glance drones, driver assistance systems and mobile medical diagnostic equipment don't appear to have much in common. But in reality they do: they all make increasing use of [image processing](#) components, for example for detecting obstacles and pedestrians. Image processing can also be used with mobile X-ray equipment to ensure adequate image quality at reduced radiation levels, thus considerably reducing radioactive exposure.

In contrast to a workstation computer, where dimensions and [energy consumption](#) are not particularly critical factors, applications like these require for small, lightweight, energy-efficient image processing components that are nevertheless real-time capable. Hardware platforms based on conventional computer architectures and processors can't properly meet these requirements. This is why embedded systems using field-programmable gate arrays (FPGAs) are often used.

Field-programmable gate arrays are logic components whose circuit structure can be freely configured using a special type of programing, usually involving the low-level language VHDL. There's a problem, however: The majority of image processing applications are written in higher-level programming languages such as C/C++, and their migration to the embedded systems is highly complicated. Not only does VHDL differ greatly from other programming languages, but the code must also be adapted to the specific hardware. This means even existing VHDL programs can't be transferred to other hardware. Software developers have to start virtually from scratch with every new system.

Starter kit for rapidly implementing especially energy-

efficient embedded systems



The stereo camera and the embedded system installed on the drone. Credit: Fraunhofer IOSB

A consortium of eight partners from six countries, including the Fraunhofer Institute for Optronics, System Technologies and Image Exploitation IOSB in Karlsruhe, has now considerably simplified this procedure in the Tulipp project. "The result is a development platform consisting of design guidelines, a configurable hardware platform and a real-time-capable operating system that supports multicore processors, as well as a programming tool chain," says Dr.-Ing. Igor Tchouchenkov, group manager at Fraunhofer IOSB. "A starter kit put on the market by one of our partners in Tulipp provides additional support. The starter kit

makes developing such applications much faster and easier. Porting C++ programs to FPGA, which frequently means several months of work for the developer, can be handled within only a few weeks using the Tulipp starter kit."

This means the developer first has to consider, based on the software programmed in C++, which code elements should be distributed to which hardware components and which program steps could be optimized or parallelized. The formulated design guidelines provide help with this task. Then the starter kit comes into play. It contains the configurable hardware to which the necessary sensors and output devices can be connected, the multiprocessor-capable real-time operating system, and what is called the STHM toolchain. The applications in the toolchain make it possible to optimize the C++ program in such a way that it can be ported to the FPGA as easily and quickly as possible. "One special focus of the toolchain is on energy optimization: after all, the aim is to design image processing systems that can be powered by a small battery whenever possible," says Tchouchenkov. "The toolchain makes it possible to individually display and optimize energy consumption for each code function."

A hundred times faster than a high-end PC

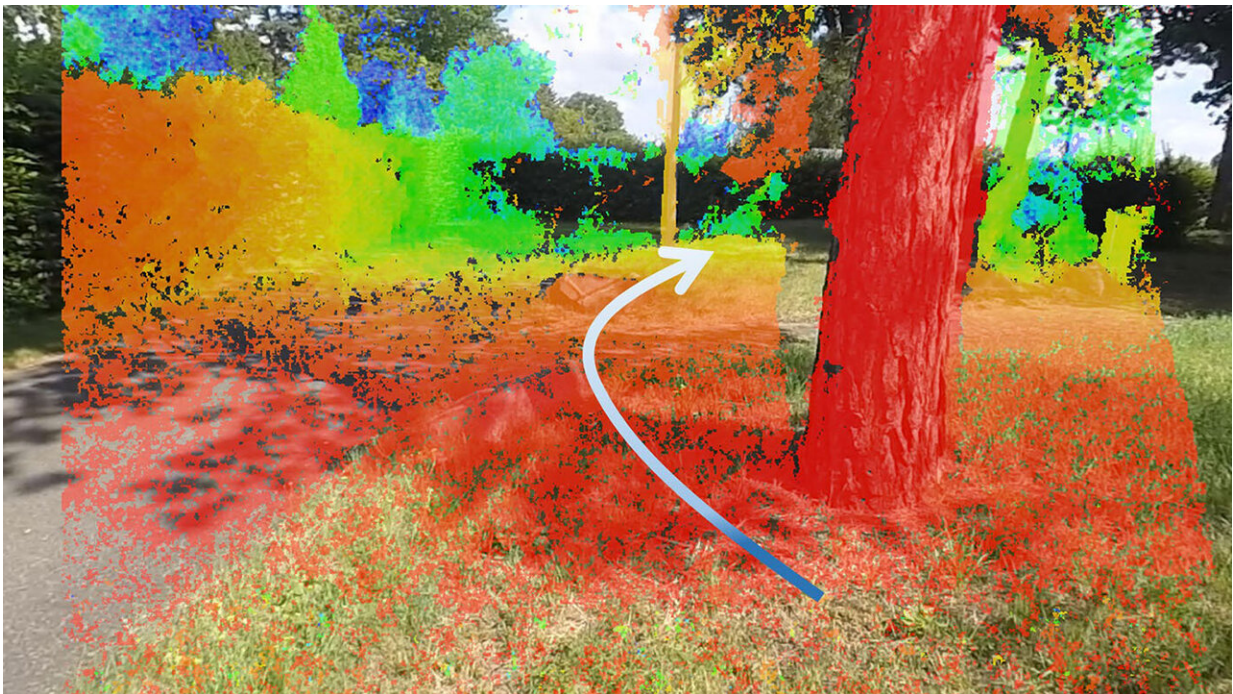


Video image from one of the drone's two cameras. Credit: Fraunhofer IOSB

The consortium worked through three specific use cases in order to develop and test the Tulipp platform: The Fraunhofer IOSB research team addressed stereo camera based obstacle detection for drones, while other project partners worked on pedestrian detection in the vicinity of a car and on live quality enhancement of X-ray images taken by mobile C-Arms during surgical operations. In Tulipp they ported the corresponding image processing software from C++ to FPGA.

The results are impressive: The processing, which originally took several seconds to analyze a single image on a high-end PC, can now run on the drone in real time, i.e. now approximately 30 images are analyzed per second. "The speed of pedestrian detection algorithm could be increased by a factor of 100: Now the system can analyze 14 images per second compared to one image every seven seconds. Enhancement of X-ray

image quality by applying noise-removing image filters allowed reducing the intensity of radiation during surgical operations to one fourth of the previous level. At the same time energy consumption could be significantly reduced for all three applications.



The same image with overlaid depth information from the stereo image analysis, highlighting close obstacles in red. The arrow indicates the resulting recommended evasive path. Credit: Fraunhofer IOSB

Provided by Fraunhofer-Gesellschaft

Citation: Developing embedded systems faster (2019, September 2) retrieved 26 April 2024 from <https://techxplore.com/news/2019-09-embedded-faster.html>

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.