



Bild 1 | Das Open Camera Concept bietet auch Einsteigern eine leistungsfähige und flexible Hardware- und Softwareplattform, um erfolgreich das 'going smart' zu meistern (v.l.): GigEPro, Corsight und iam.

Smart Vision Vielfalt

Open Camera Concept für eigene, smarte Vision-Lösungen

AUTOR: GREGOR MARTIN, MARKETING MANAGER, NET GMBH | BILDER: NET GMBH

Um individuelle Vision-Lösungen zu realisieren, kann der Anwender bei dem NET Open Camera Concept seine Algorithmen in verschiedene intelligente Kameraoptionen integrieren und jederzeit anpassen. NET stellt dafür die Hardwarebasis und Entwicklungsumgebung zur Verfügung.

Anwender haben damit die Möglichkeit zu 100% eigene Lösungen mit exakt aufeinander abgestimmten Vision Systemen und individuell kombinierbarer Software aus bestehenden Algorithmen, lizenzierbaren Bibliotheken und OpenCV zu erstellen. Der folgende Beitrag zeigt, welche Möglichkeiten sich damit für Anwender und Programmierer ergeben.

FPGA-gestützte (Vor-)Verarbeitung in der Kamera

Durch Dezentralisierung der Bildverarbeitung vom Host-PC auf kompakte Kameras vor Ort lässt sich der Bildverarbeitungsprozess für bestimmte An-

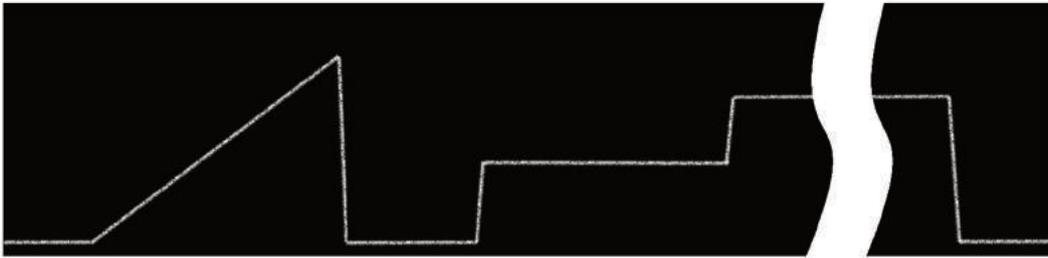
wendungen entscheidend beschleunigen. Mit der GigE Vision-Kamera GigE-Pro bietet NET z.B. eine Lösung an, die Algorithmen parallel im FPGA prozessiert. Gleichzeitig behält die Kamera ihren kompakten Formfaktor bei. Eine autarke dezentrale Bildverarbeitung, die vollständig ohne Zentralrechner auskommt, wird dabei jedoch nicht angestrebt. Ein Beispiel ist die integrierte 3D-Lasertriangulation in einer Kamera. Aus 2D-Daten werden im Zuge einer Frameanalyse 1D-Datensätze gewonnen. Die erzielte Datenreduktion in Echtzeit erlaubt die Auswertung und Übertragung deutlich größerer ROI an den Host-PC. So kann etwa mehr

Bahnware in schnelleren Tempo präzise ausgewertet und weiterverarbeitet werden. Damit Anwender ihre Algorithmen in den kamera-internen FPGA programmieren können, gib es das Open Camera Concept. Es basiert auf dem NET Framework. Die dezidierte Schnittstelle (Wrapper) für die Programmierung des VHDL-Codes wird über das SynView API von NET bereitgestellt. Der SynView Explorer bietet ein leicht bedienbares GUI zur Steuerung aller Standard-GenICam Funktionen, aber auch kundenspezifischer Funktionen. Voraussetzung zur freien Programmierung des Systems sind FPGA-Programmierkenntnisse auf Seiten des Anwenders.

Hybride dezentrale Bildverarbeitung

Mit dem Smart Vision System Corsight kann dagegen die gesamte Bildverar-

Original image acquisition sensor:



Resulting data from FPGA:

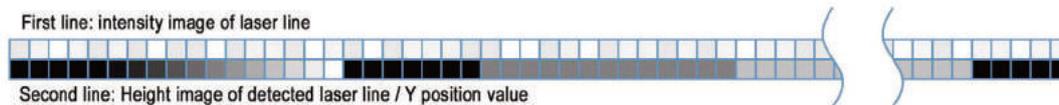


Bild 2 | Die GigEPro Kameraprozessiert Algorithmen parallel im FPGA. So können bei der 3D-Lasertriangulation aus 2D-Daten mit einer Frameanalyse 1D-Datensätze gewonnen werden.

cepts neben dem FPGA auch auf die CPU ausgeweitet. Wo PC-seitig auch workflow-basierte Softwarelösungen zur Anwendung kommen können, sind zur Programmierung des FPGA weiterhin Expertenkenntnisse gefragt. Danach lässt sich über das Framework auf den kamerainternen FPGA zugreifen und der volle Leistungsumfang des Systems nutzen.

beitungsfunktion dezentral bewältigt werden. Je nach Auslegung agiert das System sogar als Aktuator zur Steuerung der Peripherie. Dabei können die einzelnen Funktionseinheiten untereinander oder inline autark handeln. Die integrierte Recheneinheiten des Systems (Intel Quad Core-CPU und FPGA) lassen sich individuell mit Algorithmen belegen oder für Bibliotheken wie Hal-

con, MIL, VisionPro, Adaptive Vision Studio oder OpenCV nutzen. Aufgrund der x86 PC-Architektur ist die Anwendungserstellung und das Servicing in gewohnter Arbeitsumgebung (Win10, Linux) möglich. Darüber hinaus ist das System auch mit Zeilensensoren als smarte Zeilenkamera einsetzbar. Im Hinblick auf Software wird hier die Umsetzung des NET Open Camera Con-

Ease-of-Use mit SoC-Architektur

Das Smart Vision System iam ist ideal für autarke Prozessen mit individuellen, flexiblen Softwarelösungen. Dank diverser Schnittstellen (DP+, USB, Feldbus) und Optik-Mounts (C, M42, F, M12) sowie optionalen Multi-Sensoren-Adaptionen, die schrittweise in 2021/22 implementiert werden, bietet das System

Workflow der Hardwarebeschleunigung

Die Hardwarebeschleunigung mit iam aus Entwicklersicht erfolgt in einem FPGA-Modul, das über die CPU gesteuert und über das DDR-RAM mit Daten versorgt wird. Ein Zutun des Nutzers ist dabei nicht erforderlich. Der Workflow läuft wie folgt:

- Identifikation, welche Funktionen in der Bildverarbeitungskette die meiste Rechenzeit in Anspruch nehmen. Um den Flaschenhals zu finden, stellt Xilinx ein Profiling Tool zur Verfügung.
- Integration der Funktion in das FPGA: In Abhängigkeit der Expertise, Präferenz des Nutzers und der zu beschleunigenden Funktion kann eine der folgenden Varianten gewählt werden.
 - a) Xilinx OpenCV: Handelt es sich bei der zu beschleunigenden Funktion um eine OpenCV-Funktion, kann diese durch ein von Xilinx erstelltes Pendant ersetzt werden. Dabei muss der Nutzer nur geringe Änderungen am Code vornehmen und kann eine von über 50 OpenCV Funktionen nutzen, die in der Xilinx xfOpenCV Library zur Verfügung stehen.
 - b) C++: Eine weitere Möglichkeit der Funktionsbeschleunigung im FPGA – ohne Erfahrung mit VHDL oder Verilog – ist die Verwendung von C++ in Verbindung mit Pragmas. Die Xilinx Tools (Vitis, Vivado) interpretieren dann den in C++ geschriebenen Beschleuniger und generieren daraus ein FPGA-Modul.
 - c) RTL: Ist VHDL/Verilog Expertise vorhanden, kann der Nutzer auch direkt seine Funktion umsetzen, z.B. als VHDL-Modul.
- Mit nur geringen Anpassungen der Software-Applikation wird die rechenintensive Funktion nun durch das FPGA-Modul ersetzt.

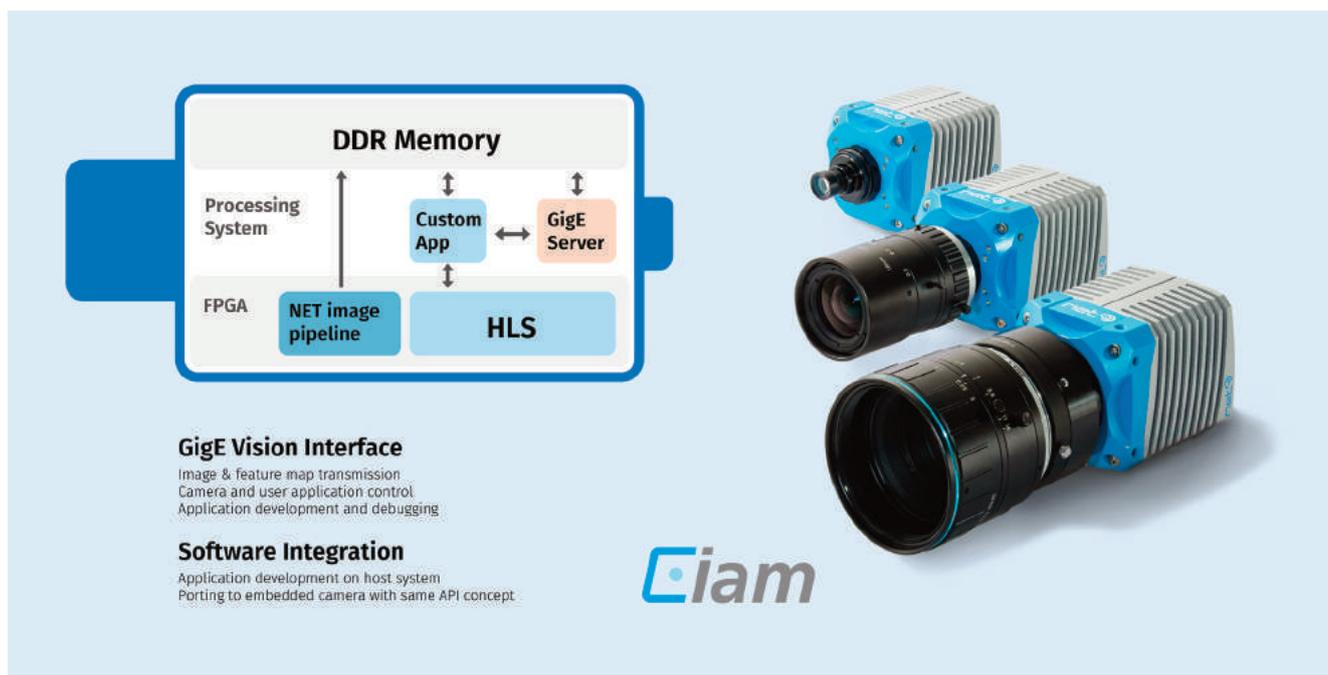


Bild 3 | Die Recheneinheiten von iam (ARM-CPU und FPGA) lassen sich individuell mit Algorithmen belegen oder für verschiedene Bibliotheken nutzen.

einen größeren Funktionsumfang als Corsight. Unterschiedlich ist, wie Anwendungen im System realisiert werden. Applikationen, die den FPGA nutzen, können mit iam in C und C++ erstellt werden. Dies ermöglicht es Softwareentwicklern ohne VHDL-Know-how das System zu nutzen. Durch die Verwendung von Standards wie OpenCL und Vitis zur automatischen Erzeugung von FPGA-Binärcode und Linux als Betriebssystem gelingt der Umstieg von PC-basierten Vision-Systemen auf iam problemlos. Eine kostenlose Hilfe für Einsteiger ist zudem die Online-Community mit zahlreichen Tutorials, die das Arbeiten mit der Vitis Softwareplattform von Xilinx erleichtern. Zusätzlich haben Lösungsanbieter die Option einen erfahrenen Systempartner mit der Programmierung zu beauftragen.

Kern von iam ist die System-on-Chip Prozessor-Architektur. Dank der hybriden Plattform können in diesem System Visionfunktionen zwischen ARM-Prozessor und FPGA effizienter hardwarebeschleunigt werden, als es durch den User manuell möglich wäre. Der resultierende Effizienzsprung sorgt für Anwendungen ohne Latenz, bezogen auf den Frametakt. Große Praxisrelevanz erlangt die Hardwarebeschleunigung bei komplexeren Berechnungen. Wenn beispiels-

weise externe Libraries oder KI eingesetzt werden, kann iam bereits fertige Operationen (Bildvergleich, Blob-Analyse, Farberkennung usw.) durch den SoC-bedingten wechselseitigen Speicherzugriff performanter lösen.

Fazit

Das Open Camera Concept bietet auch Einsteigern eine leistungsfähige und flexible Hardware- und Softwareplattform, um erfolgreich das 'going smart' zu meistern. Weitere Vorteile sind das breite Sensor- und Optikportfolio in Verbindung mit den optionalen Customizing-Optionen. Mit dem hardwarebeschleunigten Smart Vision System iam kann der Anwender ein Maximum an Perfor-

mance herausholen, ohne dass er seine Arbeitsweise oder Software grundlegend ändern muss. ■

www.net-gmbh.com

- Anzeige -