

Bildentzerrung für die leanXcam

Signalverarbeitung / Prof. Dr. Josef Goette

Experte: Felix Kunz, DIGITAL-LOGIC AG

In der Bildverarbeitung, wie zum Beispiel in der Stereo Vision, gehen viele Verarbeitungsalgorithmen von unverzerrten Bildern aus. In der Realität weist jedoch jede Kamera und insbesondere jede Optik vielfältige Verzerrungen auf. Zu nennen sind neben anderen die *radial distortion* (Fischaug-Effekt), nichtlineare optische Verzerrungen und Einflüsse der Kamerageometrie und Position. Diese Gründe erfordern Algorithmen zur Bildentzerrung, zur Wiederherstellung der realen Geometrie eines Objektes. Solche Algorithmen müssen mit vernünftigen Aufwand oder idealerweise sogar automatisch trainierbar sein, da jede Kamera ein eigenes Set von Verzerrungsparametern aufweist.



Michael Gasser

Ausgangslage

Die leanXcam ist von der Firma Supercomputing Systems AG (SCS) als low cost embedded smart camera konzipiert worden. Die Kamera enthält einen Blackfin-DSP-Prozessor und läuft unter dem frei erhältlichen Betriebssystem μ CLinux. Der Kamerasensor selber liefert Bilder in der Auflösung von 752 x 480 Bildpunkten mit maximal 60 Bildern pro Sekunde. Die Kommunikation mit der Aussenwelt funktioniert über Ethernet.

Zur Entzerrung der Kamera bei kommerziellen Anwendungen von SCS werden Testbilder im Einsatz erstellt; anschliessend berechnet Matlab die Entzerrungsparameter offline. Dieses Vorgehen ist umständlich und soll durch eine Erweiterung der leanXcam Library ersetzt werden, sodass die Bildentzerrungsparameter direkt auf der Kamera berechnet werden können.

Realisierung

Im ersten Teil der Arbeit wurde ein Vergleich von zwei verschiedenen Entzerrungsalgorithmen durchgeführt. Als Ausgangspunkt diente der bereits existierende Algorithmus der Firma SCS. Als Alternative wurde der Algorithmus von Zhengyou Zhang analysiert, der in der OpenCV-Library (open source computer vision library) von Intel implementiert ist. Diese Library ist



für die Programmiersprachen C und C++ geschrieben und enthält Algorithmen für die Bildverarbeitung.

Aufgrund des Vergleichs der beiden Algorithmen bezüglich Qualität und Umsetzbarkeit und in Betracht, dass in OpenCV bereits eine C-Implementierung existiert, wurde für die vorliegende Implementierung der Zhang-OpenCV-Algorithmus ausgewählt.

Mittels Anpassungen im OpenCV-Sourcecode und den entsprechenden Angaben für den Blackfin Cross-Compiler kann die Library kompiliert und in den μ CLinux-Kernel integriert werden. OpenCV verwendet für die Ka-



Testobjekt für die Kalibrierung

librierung ein Testbild, welches zuerst aufgenommen wird; anschliessend werden daraus die Entzerrungsparameter berechnet. Die Bedienung des Kalibrier- und Entzerrungsablaufs erfolgt durch ein Webinterface.

Perspektiven

Die OpenCV-Bibliothek verwendet Floating-Point Arithmetik, wofür der Blackfin-Prozessor keine Hardware-Unterstützung bietet. Damit die Programme nicht zu langsam rechnen, müssen sie für den Blackfin optimiert werden. Der Blackfin-Cross-Compiler bietet eine Soft-Floating-Point-Library an, um Floating-Point-Arithmetik zu emulieren. Mit deren Verwendung liegt die Ausführungszeit des Programms in einem angemessenen Rahmen. Um OpenCV noch besser für den Blackfin zu adaptieren, könnten in Zukunft die relevanten Funktionen in Fixed-Point-Arithmetik umgeschrieben werden.