

Eine intelligente Kamera zur Verkehrsüberwachung

Echtzeitverarbeitung und Datenvolumen stellen hohe Anforderungen an Hardware und Software

Im Transport- und Verkehrsbereich treten besonders hohe Anforderungen an Sicherheit und Auslastung auf. Daraus ergeben sich neue Ansätze in der Verkehrsüberwachung und Verkehrsleitung. Ein sehr gutes Beispiel, an dem sich dieser Trend illustrieren lässt, stellen *intelligente Kameras* zur Verkehrsüberwachung dar. Neben der Aufnahme der optischen Information und der Umwandlung dieser Information in elektrische Signale führen intelligente Kameras auch Bildverarbeitung (Videokompression und Videoanalyse) und Netzwerkkommunikation durch. Intelligente Kameras vereinen daher Bildaufnahme mit Bildverarbeitung und Kommunikation in einem einzelnen eingebetteten Gerät.

Der Begriff der intelligenten Kamera wurde unter anderem von Wayne Wolf¹⁾ geprägt. Übernehmen mehrere intelligente, mit digitalen Signalprozessoren ausgestattete Kameras die Erkennung von menschlichen Bewegungen und Gesten, können intelligente Räume («smart

Bernhard Rinner, Horst Bischof, Michael Bramberger, Andreas Doblander, Arnold Maier, Roman Pflugfelder, Helmut Schwabach

rooms») realisiert werden, in denen die Kameras vordefinierte Aktionen, wie beispielsweise die Steuerung einer Präsentation oder eines Fernsehapparats, auslösen [1]. Einsätze von Kamera-Arrays zur Entwicklung solcher intelligenten Umgebungen wurden von Jeremy R. Cooperstock²⁾ erforscht. Beispielsweise, indem eine grosse Anzahl billiger Kameras mit eingebetteter Intelligenz verwendet wurde, um Personen und Objekte in Räumen, Gebäuden und auf öffentlichen Plätzen zu verfolgen [2].

Aber auch im Bereich der verteilten Überwachung gewinnen intelligente Kameras mehr und mehr an Bedeutung.

Carlo Regazzoni³⁾ etwa untersucht die Möglichkeiten verteilter Systeme mit intelligenten Kameras, welche in Gruppen an einer gemeinsamen Lösung von komplexen Problemen arbeiten [3]. Die Schwerpunkte der Arbeiten liegen dabei im Zusammenführen der einzelnen Informationen und in der Verfolgung von Personen und Objekten auf öffentlichen Plätzen. Ferner werden die Lastverteilung zwischen den einzelnen Kameras und so genannte intelligente Hubs in diesem Projekt untersucht. Im Rahmen des VSAM-Projektes (Video Surveillance and Monitoring [4]) wurden unterschiedliche Sensoren in einer Stückzahl von über 1000 dazu benutzt, Bewegungen von Objekten, Fahrzeugen und Personen auf einem Freigelände zu überwachen und zu verfolgen. Auch hier verfügen alle Sensoren über eine eingebettete Intelligenz und eine Netzwerkschnittstelle zum Austausch von Daten und Befehlen.

Im Rahmen eines gemeinsamen Forschungsprojektes der Technischen Universität Graz (TUG) und des Austrian Research Centers Seibersdorf (ARCS) wird eine intelligente Kamera zur Verkehrsüberwachung (SmartCam [5]) entwickelt⁴⁾. Diese intelligente Kamera erfasst

einen Videostrom, analysiert und komprimiert die Videodaten und transferiert schliesslich die komprimierten Videodaten gemeinsam mit den Analyseergebnissen über ein Netzwerk zu einer Überwachungszentrale. Die Videoanalyse in der Kamera bestimmt quantitative Verkehrsparameter, wie beispielsweise die Durchschnittsgeschwindigkeit der Fahrzeuge und die Auslastung der Fahrstreifen, aber auch qualitative Aussagen, wie etwa automatische Stillstands- und Raucherkennung.

Anforderungen an intelligente Kameras

Anwendungen im Bereich der Verkehrsüberwachung stellen hohe Anforderungen an die Hard- und Software von intelligenten Kameras. Erstens müssen die Videodaten eine hohe örtliche und zeitliche Auflösung besitzen und in Echtzeit verarbeitet werden. Viele herkömmliche Bildsensoren bieten jedoch nur kleine Bildformate wie z.B. CIF⁵⁾ und QCIF⁶⁾ an. Überwachungsanwendungen erfordern jedoch höhere Auflösungen (z.B. PAL⁷⁾ mit 720×576 Pixeln) und Bildraten von bis zu 25 fps⁸⁾. Zweitens muss die intelligente Kamera eine sehr hohe Rechenleistung bieten, um die anspruchsvollen Verarbeitungsschritte zeitgerecht ausführen zu können. Die wichtigsten Schritte beinhalten

- Videokompression nach MPEG-4⁹⁾
- Videoanalyse zur Verkehrsüberwachung
- Berechnung von Verkehrsparametern
- Kamerasteuerung und Firmware.

Drittens müssen die komprimierten Videodaten und die berechneten Verkehrsparameter zur Überwachungszentrale gesendet werden. Um eine flexible und fehlertolerante Kommunikation gewährleisten zu können, sollen unterschiedliche Verbindungen wie beispielsweise über Ethernet, Wireless-LAN und GSM/GPRS¹⁰⁾ möglich sein. Schliesslich stellen auch niedriger Energieverbrauch und Echtzeitfähigkeit wichtige Systemanforderungen der intelligenten Kamera dar.

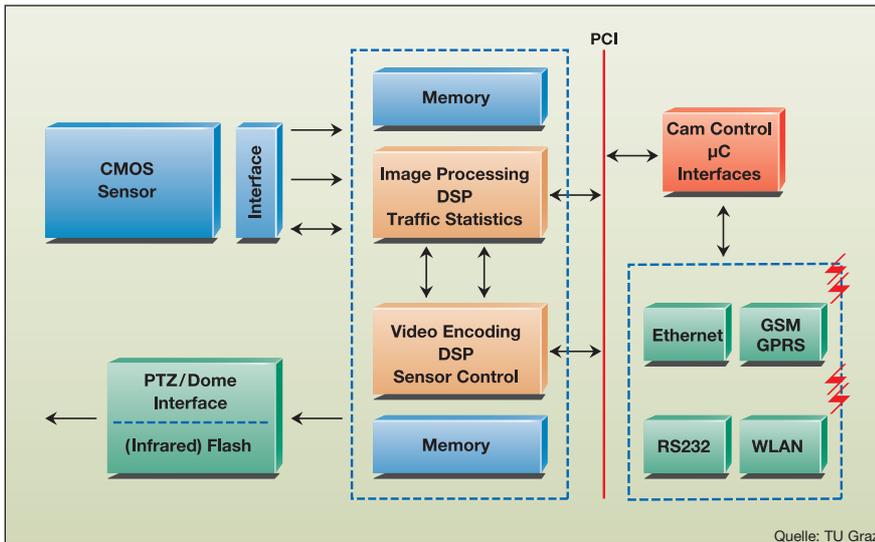


Bild 1 Gesamte Systemarchitektur der intelligenten Kamera

Architektur der intelligenten Kamera

Wie in Bild 1 dargestellt, ist die Architektur der in diesem Beitrag vorgestellten intelligenten Kamera aus drei wesentlichen Einheiten aufgebaut:

- dem Videosensor
- der Verarbeitungseinheit
- der Übertragungseinheit.

Für die Bilderfassung wird ein CMOS¹¹⁾-Sensor eingesetzt, da diese Sensoren am besten den Anforderungen einer intelligenten Kamera für die Verkehrsüberwachung gerecht werden. CMOS-Sensoren zeichnet besonders die hohe Dynamik auf Grund ihrer logarithmischen Charakteristik aus. Weiters bieten viele CMOS-Sensoren bereits on-chip ADC¹²⁾ und Verstärker; dies vereinfacht den Schaltungsaufbau.

Die Videodaten werden vom Sensor an die Verarbeitungseinheit weitergereicht. Auf Grund der aufwendigen On-Board-Bildverarbeitungsalgorithmen sind die Anforderungen an die Rechenleistung sehr hoch. Eine grobe Abschätzung für die Videokompression und -analyse ergibt eine benötigte Rechenleistung von etwa 10 GIPS¹³⁾. Digitale Signalprozessoren (DSP) erfüllen am besten die hohen Anforderungen an die Rechenleistung sowie die Einschränkungen auf Grund der Implementierung als eingebettetes System (Energieverbrauch). Die SmartCam ist mit zwei der derzeit leistungsfähigsten DSP (TMS320C64x von Texas Instruments) ausgestattet, wobei ein DSP beinahe ausschließlich für die MPEG-4-Videokompression eingesetzt wird; der zweite DSP übernimmt die Aufgaben der

Videoanalyse. Um das Timing zu vereinfachen, ist der CMOS-Sensor über einen FIFO-Speicher¹⁴⁾ mit einem DSP verbunden. Die einzelnen Bilder werden vom FIFO-Speicher dann in den externen Speicher des DSP transferiert. Die DSP sind über PCI¹⁵⁾ mit einem Netzwerk-Controller verbunden. Dieser steuert die gesamte Kamera und ist auch für die Weiterleitung der Daten an die Kommunikationseinheit verantwortlich.

Der Netzwerk-Controller bietet bereits viele Schnittstellen, wie beispielsweise USB, Ethernet und RS232, an. Für diese Schnittstellen müssen dann lediglich die physikalischen Schichten in der Kommunikationseinheit realisiert werden. Der Datenaustausch zwischen Kommunikationseinheit und Netzwerk-Controller ist über eine generische Schnittstelle implementiert. Damit können einfach zusätzliche Netzwerkverbindungen beispielsweise über GSM/GPRS und Wireless-LAN aufgebaut werden. In der Kommunikationseinheit befinden sich auch serielle Schnittstellen (RS232 und RS422) und digitale Ausgänge. Damit können unter anderem die Kamerapositionierung (PTZ¹⁶⁾, dome¹⁷⁾ und die Blitzlichteinheit gesteuert werden.

Geringer Energiebedarf stellt ein wichtiges Kriterium unserer intelligenten Kamera dar. Neben der Verwendung von «energieschonenden» Komponenten in der Kamera wird mit Hilfe von *dynamischem Power Management* (DPM) zusätzlich versucht, den Energieverbrauch im Betrieb möglichst gering zu halten. Um DPM einsetzen zu können, müssen sich die einzelnen Komponenten im Betrieb in verschiedene Power Modes um-

schalten lassen. Jeder Power Mode ist durch unterschiedlichen Funktionsumfang und Energieverbrauch der Komponenten charakterisiert. Die grundlegende Idee von DPM ist nun, die einzelnen Komponenten in jenen Power Mode zu schalten, der die erforderliche Funktionalität (und Rechenleistung) gerade gewährleisten kann, dabei aber den Energieverbrauch minimiert. Die Umschaltung der einzelnen Komponenten erfolgt meist über einen zentralen Power Manager, der als Teil der Firmware im Netzwerk-Controller realisiert ist. Viele Komponenten der intelligenten Kamera können über den I2C-Bus¹⁸⁾ konfiguriert werden.

Software

Eine besondere Herausforderung stellt die Implementierung der Software auf der intelligenten Kamera dar. Einerseits soll die Softwarearchitektur flexibel und erweiterbar sein, um auf die Änderungen in den Überwachungsanwendungen rasch reagieren zu können. Andererseits muss auf eine hohe Leistungsfähigkeit (Rechenleistung) der implementierten Algorithmen auf der eingebetteten Plattform geachtet werden.

Architektur der eingebetteten Software

Die Funktionalität und Komplexität der SmartCam erfordern eine offene und flexible Architektur der eingebetteten Software. Dies gilt insbesondere für die Software der DSP. Videokompression, Videoanalyse und Firmware werden daher nach dem Algorithmus-Standard XDAIS und dem «Reference Framework» (RF) [6] von Texas Instruments implementiert. Das RF kann als Middleware für eingebettete Software betrachtet werden und übernimmt gemeinsam mit einem Betriebssystem (DSP/BIOS¹⁹⁾) Ressourcenmanagement und Kommunikation. Softwaremodule, wie z.B. die Videokompression, können einfach geändert oder hinzugefügt werden. Viele Hersteller bieten bereits Module für das RF an, die dann leicht in unsere Kamera integriert werden können.

Effiziente Implementierung der Module

Software für Video- und Bildverarbeitung wird meist auf Desktop-Computern in mächtigen Hochsprachen (C++ bzw. MATLAB) entwickelt. Es sind bereits Werkzeuge und Bibliotheken für die Abbildung der in der Hochsprache entwickelten Algorithmen auf die eingebettete Plattform verfügbar. Diese Werkzeuge

unterstützen zwar recht gut das Rapid Prototyping auf der Zielplattform, die geforderte Rechenleistung der Algorithmen wird häufig jedoch nur mit Hilfe eines «Redesigns» bzw. einer teilweisen Implementierung in Maschinsprache erreicht [7]. Nachfolgend soll die effiziente Implementierung für eine eingebettete DSP-Plattform anhand eines Algorithmus zur Videoanalyse – der automatischen Erkennung von stationären Fahrzeugen – demonstriert werden.

Automatische Stillstandserkennung (ASE)

Ein Verfahren zur automatischen Erkennung von stationären Fahrzeugen in Tunnels wurde von Pflugfelder et al. [8] entwickelt. Es beruht auf der Annahme, dass die Lichtverhältnisse in Tunnels relativ konstant und die Überwachungskameras fix montiert sind. Veränderungen der Intensität der aufgenommenen Bilder werden unter diesen Annahmen daher nur durch die Bewegung der Fahrzeuge und Rauschen (Lichtkegel, Reflexionen) verursacht.

Genau diese Intensitätsschwankungen werden beim ASE-Verfahren statistisch ausgewertet. Grauwertbilder mit 8-Bit-Auflösung dienen als Eingabe; der Algorithmus liefert jenen Bildbereich, in welchem ein stationäres Fahrzeug erkannt wurde, als Ausgabe. Ein statistisches Modell des Szenenhintergrunds bildet die Basis des ASE-Verfahrens. In diesem Hintergrundmodell werden jene Bildbereiche zusammengefasst, in denen in der jüngsten Zeit keine grossen Intensitätsschwankungen aufgetreten sind. Bildbereiche (Pixel), die nicht im Hintergrundmodell inkludiert sind, können daher dem Bildvordergrund zugeordnet werden. Mit jedem neuen Bild, das von der Kamera geliefert wird, wird nun das Hintergrundmodell neu berechnet (adaptiert). Ein stationäres Fahrzeug wird dann erkannt, wenn ein genügend grosser Bereich des Vordergrunds als (neuer) Hintergrund identifiziert wird. Die Adaptierung des Hintergrundmodells sowie die Erkennung einer (langfristigen) Intensitätsveränderung erfolgen mit Hilfe von statistischen Parametern der einzelnen Pixel. Die wichtigsten Schritte des ASE-Verfahrens sind in Bild 2 dargestellt.

Abbildung des Algorithmus auf DSP-Plattform

Der ASE-Algorithmus wurde mit Hilfe von MATLAB entwickelt und getestet. Eine direkte Abbildung des Algorithmus auf die eingebettete DSP-Plattform führt jedoch meist zu einem unbefriedigenden Ergebnis [7]. Die schlechte Leistung wird

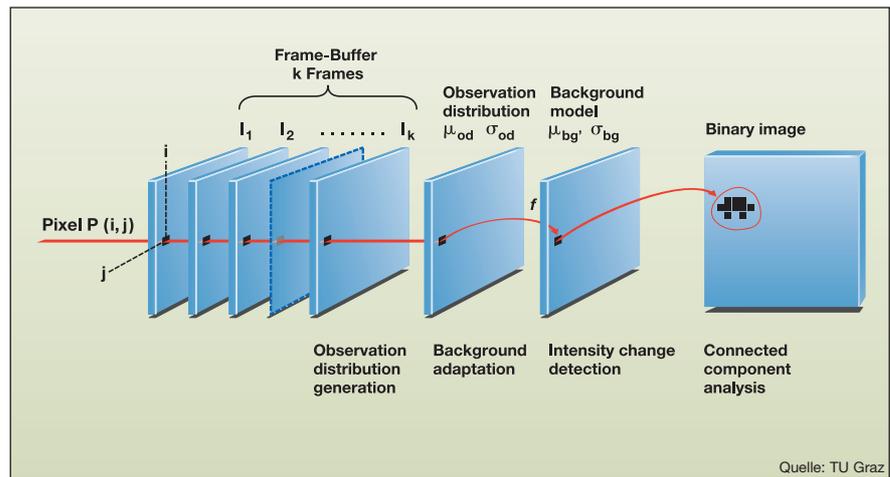


Bild 2 Verarbeitungsschritte der automatischen Stillstandserkennung (ASE)

hauptsächlich durch folgende Gründe verursacht:

- *Externe Speicherzugriffe:* Viele Hochsprachen unterstützen den Einsatz von komplexen und grossen Datenstrukturen. Zusätzlich wird die Speicherverwaltung – bewusst – vor dem Programmierer verborgen gehalten, was zu hohem Speichertransfervolumen führt. Eingebettete DSP-Plattformen besitzen meist nur sehr begrenzten (internen) Speicher. Exzessiver Zugriff auf externe Speicherbereiche reduziert daher stark die erzielbare Rechenleistung.
- *Datentransfer:* Die meisten eingebetteten Prozessoren bieten Caches und DMA²⁰, um den Datentransfer zwischen externem Speicher und Prozessor zu erhöhen. Caches sind für den Programmierer transparent, erzielen jedoch in der Regel einen geringeren Leistungsgewinn als DMA-Transfers. Für den effizienten Einsatz von DMA muss das Datenzugriffsmuster jedoch bekannt und regelmässig sein. Dies trifft für viele Algorithmen der Video- und Bildverarbeitung zu.
- *Zahlenformat:* Da im Desktop-Bereich das verwendete Zahlenformat meist nur eine untergeordnete Rolle spielt, wird häufig die Fließkomma-Darstellung verwendet. In eingebetteten Plattformen finden sich jedoch meist nur Fixkomma-Prozessoren, auf denen Fließkomma-Operationen aufwendig in Software realisiert werden müssen.

Für die optimale Abbildung des ASE-Algorithmus auf die DSP-Plattform wurden diese Gründe berücksichtigt [9]. Die wichtigsten Optimierungsschritte sind nachfolgend aufgeführt.

- *Pixel-orientierte Berechnung:* In der MATLAB-Implementierung werden die einzelnen Operationen auf komplette Matrizen (Bilder) angewendet. Um den Speichertransfer zu reduzieren, wurden diese Operationen von einer «Bild-orientierten» auf eine «Pixel-orientierte» Verarbeitung umgestellt. Damit muss die Bildmatrix nur einmal vom externen Speicher geladen werden; Zwischenergebnisse der Verarbeitungskette können nun im internen Speicher oder in Registern gehalten werden.
- *DMA-Transfer mit Doppel-Bufferung:* Teilbereiche der Bilder werden mittels DMA vom externen in den internen Speicher transferiert. Dort werden die Daten verarbeitet und das Ergebnis wird dann wiederum per DMA in den externen Speicher geschrieben. Der Datentransfer und die interne Verarbeitung in der CPU werden mit Hilfe von zwei Bufferpaaren entkoppelt. Dabei wird ein Bufferpaar – bestehend aus Eingabe- und Ausgabebuffer – für den DMA-Transfer verwendet; das andere Bufferpaar stellt zur gleichen Zeit Ein- und Ausgabe für die Verarbeitung dar. Für die Verarbeitung bzw. den Transfer des nächsten Bildbereiches werden dann die Bufferpaare vertauscht (Ping-Pong Buffering).
- *Fixkomma-Zahlenformat:* Der MATLAB-ASE-Algorithmus wurde in eine Fixkomma-Verarbeitung im Q9.6-Format²¹ umgewandelt. Neben der direkten Ausführung der Fixkomma-Operationen im DSP ergeben sich daraus noch zwei weitere Vorteile. Erstens wird der Speicherbedarf reduziert (16-Bit-Fixkomma-Zahlenformat statt 32-Bit-Fließkomma-Zahlenformat). Zweitens kann die Parallelverarbei-

Variante	Beschreibung	Zyklen [M]	Zeit [ms]	Bildrate
1	MATLAB, 2,4-GHz-PC		2900	0,34
2	C++, 600-MHz-DSP	4200	7000	0,14
3a	C, DSP (ohne DMA)	190	317	3,15
3b	C, DSP (mit DMA)	152	254	3,90
3c	C, DSP (mit DMA und «packed data»)	125	208	4,80

Ausführungszeit und erzielte Bildrate verschiedener ASE-Implementierungen

tung im DSP erhöht werden, da einige 32-Bit-DSP zwei 16-Bit-Zahlen bzw. vier 8-Bit-Zahlen parallel in einer Verarbeitungseinheit berechnen können (Packed-Data Processing).

Implementierungsvarianten

Die Tabelle fasst die Ergebnisse mehrerer Implementierungsvarianten des ASE-Algorithmus zusammen [9]. Ausgangspunkt ist die MATLAB-Implementierung (Variante 1 in der Tabelle), die auf einem Standard-PC (Pentium 4 mit 2,4 GHz) 2,9 Sekunden für die Berechnung eines Bildes in PAL-Auflösung benötigt. Es wurden zwei unterschiedliche Versionen des ASE-Algorithmus auf dem DSP realisiert. In der ersten Version wurde der MATLAB-Code direkt in eine klassenbasierte C++-Implementierung portiert. Die Bild-orientierte Verarbeitung und die Fliesskomma-Verarbeitung wurden dabei beibehalten. Die erzielte Leistung auf einem 600-MHz-TMS320C6416-DSP war unbefriedigend (Variante 2). In der zweiten Version wurden die weiter vorne bereits erwähnten Optimierungsschritte berücksichtigt. Damit liess sich ein mehr als 10facher Leistungsgewinn erreichen. In der Variante 3a wurden die Pixel-orientierte Verarbeitung und das Festkomma-Zahlenformat umgesetzt und in der Variante 3b Caching durch Ping-Pong Buffering mittels DMA ersetzt. Variante 3c nutzt zusätzlich noch die Möglichkeit zur Parallelverarbeitung zweier 16-Bit-Zahlen aus. Alle Varianten der zweiten Version wurden in C (teilweise mit «Intrinsic»-Anweisungen) implementiert.

MPEG-4-Videokompression

Die Übertragung eines Videos in PAL-Auflösung mit einer Bildrate von 25 fps erfordert eine Bandbreite von rund 20 MB/s (im Videoformat YUV 4:2:0 mit 8 Bit pro Pixel). Um Bandbreite zu sparen, werden die Videodaten in der SmartCam komprimiert. Der MPEG-4-Standard bietet sich zur Videokompression für die Verkehrsüberwachung an, da auch bei hohen Kompressionsraten noch eine ent-

sprechende Bildqualität erreicht werden kann. Ausserdem lassen sich Kompressionsrate und Bildqualität sehr gut parametrieren und können daher leicht an die wechselnden Anforderungen in der Verkehrsüberwachung (beispielsweise nach einem Unfall) angepasst werden.

In der SmartCam wird ein kommerzieller TMS320C64x-optimierter MPEG-4-Kodierer eingesetzt. Auf Grund der hohen Anforderungen an die Rechenleistung (etwa 4000 MIPS für PAL-Auflösung mit 25 fps mit «Simple Profile»-Kodierung) wird die MPEG-4-Kodierung auf einem eigenen DSP durchgeführt.

Prototyp der intelligenten Kamera

Zum Test der Funktionalität und Leistung der SmartCam wurde ein Prototyp entwickelt. Er wurde als eingebettetes System realisiert und verfügt über alle wesentlichen Funktionen der Kamera (Bild 3). Herzstück dieses Prototyps ist das Network Video Development Kit (NVDK) von ATEME²²⁾, bestehend aus einem TMS320C6416-DSP mit 264 MB externem Speicher und einer Ethernet-Erweiterungskarte. Diese Prototypen-Plattform wurde mit einem CMOS-Sensor (LM-9618 von National Semiconductors²³⁾ erweitert, der über einen FIFO-Speicher mit dem DSP gekoppelt ist.

Obwohl der Prototyp nur mit einem DSP ausgestattet ist, können damit bereits viele Tests bezüglich der Leistung der Videokompression und Bildverarbei-



Bild 3 Prototyp der intelligenten Kamera, bestehend aus CMOS-Sensor, TMS320C64x-DSP-Plattform und einer aufgesteckten Ethernet-Netzwerkkarte

tung durchgeführt werden. Ein eingebetteter Demonstrator für die automatische Stillstandserkennung (inklusive Ethernet Streaming) wurde beispielsweise auf der Prototypen-Plattform bereits realisiert. Anhand dieses Demonstrators kann die ASE bezüglich Funktionalität, Erkennungsrate, zeitlichem Verhalten und Rechenleistung sehr gut evaluiert werden. Das Videosignal kann entweder vom On-Board-CMOS-Sensor oder von einem Videorekorder (über einen externen Videoeingang) geliefert werden. Die Evaluierung erleichtert sich dadurch wesentlich.

Ausblick

Intelligente Kameras vereinen Sensorik, Datenverarbeitung und Kommunikation als eingebettetes System und stellen eine wesentliche Komponente von Überwachungssystemen der 3. Generation dar [3]. Die Überwachungssysteme dieser jüngsten Generation werden als «vollständig digitale» Lösung realisiert, d.h., bereits beim Sensor werden die Daten digitalisiert und dann im gesamten Überwachungssystem nur noch digital verarbeitet und übertragen. Durch die Verlagerung der Datenverarbeitung von der Überwachungszentrale in die einzelnen Sensoren sowie durch alternative Kommunikationswege erhöhen sich Fehlertoleranz und Robustheit des Gesamtsystems. Ausserdem ermöglicht die nun zur Verfügung stehende Rechenleistung neue Funktionalitäten (z.B. automatische Videoanalyse), die den Benutzer vor allem bei Routinetätigkeiten unterstützen.

Die Verwendung von intelligenten Kameras beschränkt sich nicht nur auf den Bereich der Verkehrsüberwachung. Auf Grund ihrer Flexibilität können intelligente Kameras selbstverständlich auch im gesamten Bereich der Überwachungs- und Sicherheitstechnik, beispielsweise zur Überwachung von Gebäuden und öffentlichen Plätzen, eingesetzt werden. Es sind aber auch viele Anwendungen im Consumer-Bereich bzw. im Bereich «Wearable and Pervasive Computing» realisierbar. Daher ergeben sich für intelligente Kameras nahezu ungeahnte Wachstums- und Marktpotenziale.

Referenzen

- [1] W. Wolf, B. Ozerm T. Lv: Smart Cameras as Embedded Systems. IEEE Computer 35(9), pp. 48-53, 2002.
- [2] J. R. Cooperstock: Classroom of the Future: Enhancing Education through Augmented Reality. Proc. Conference on Human-Computer Interaction (HCI International 2001), New Orleans, 2001.

- [3] C. Regazzoni, V. Ramesh, G.L. Foresti (ed.): Video Communications, Processing, and Understanding for Third Generation Surveillance Systems. Proceedings of the IEEE, 89(10), Oct. 2001.
- [4] T. Kanade, R.T. Collins, A.J. Lipton, P. Burt, L. Wixson: Advances in Cooperative Multi-Sensor Video Surveillance. DARPA Image Understanding Workshop (IUW), Monterey, CA, 3-24, November 1998.
- [5] M. Bramberger, R. Pflugfelder, B. Rinner, H. Schwabach, B. Strobl: Intelligent Traffic Video Sensor: Architecture and Applications. In Workshop on Wearable Computing, Telecommunications and Mobile Computing, Graz 2003.
- [6] S. Blonstein: Reference Frameworks for eXpressDSP Software: A White Paper. Texas Instruments (SPRA094A), 2002.
- [7] K. Karadyi, V. Markandey, J. Golston, R.J. Gove, Y. Kim: Strategies for mapping algorithms to mediaprocessors for high performance. IEEE Micro 23(4), pp. 58-70, 2003.
- [8] R. P. Pflugfelder, H. Bischof: Learning spatiotemporal traffic behaviour and traffic patterns for unusual event detection. In 26th Workshop of the Austrian Association for Pattern Recognition, pp. 125-133, 2003.
- [9] M. Bramberger, J. Brunner, B. Rinner: Mapping High-Level Image Analysis Algorithms to Embedded DSP Architectures. Technical Report 05-03, Institute for Technical Informatics, Graz University of Technology, 2003.

Angaben zu den Autoren

Dr. **Bernhard Rinner** arbeitet derzeit als ausserordentlicher Professor am Institut für Technische Informatik der TU Graz, wo er 1993 das Diplom- und 1996 das Doktoratsstudium der Telematik abschloss. 1995 und 1998/99 forschte er an der University of Texas at Austin im Bereich Fehlererkennung in technischen Systemen, und im Jahr 2002 habilitierte er sich im Fach «eingebettete Systeme und Rechnerarchitekturen». Seine Forschungsthemen reichen von der Parallelverarbeitung über den Entwurf eingebetteter Systeme bis hin zu Anwendungen im Bereich digitaler Signalprozessoren, Multimedia und Echtzeitsysteme.
Institut für Technische Informatik, TU Graz, A-8010 Graz, b.rinner@computer.org

Prof. Dr. **Horst Bischof** absolvierte das Informatikstudium an der TU Wien, wo er 1993 das Doktorat und 1998 die Habilitation erlangte. In der Zeit von 1991 bis 2001 arbeitete er an der Abteilung für Mustererkennung und Bildverarbeitung der TU Wien. Seit 2001 ist er am Institut für Maschinelles Sehen und Darstellen der TU Graz als Professor für Computer Vision tätig. Die Forschungsthemen von Horst Bischof reichen von der Objekterkennung und statistischen Methoden der Mustererkennung bis hin zu Anwendungen im Bereich der Überwachung, Biometrie, Robotik und medizinischen Bildverarbeitung, wo er mehr als 160 Arbeiten in Journalen, Büchern und Konferenzen verfasst hat.
Institut für Maschinelles Sehen und Darstellen, TU Graz, A-8010 Graz, bischof@icg.tu-graz.ac.at

Dipl. Ing. **Michael Bramberger** arbeitet seit September 2002 als Forschungsassistent am Institut für Technische Informatik an der Technischen Universität Graz. Er schloss das Studium der Telematik an der TU Graz im Januar 2002 ab. Nun beschäftigt er sich im Rahmen seiner Dissertation mit intelligenten, verteilten Überwachungssystemen.
Institut für Technische Informatik, TU Graz, A-8010 Graz, bramberger@iti.tugraz.at

Dipl. Ing. **Andreas Doblender** hat das Diplomstudium der Elektrotechnik mit Schwerpunkt Informationstechnik an der Technischen Universität Graz abgeschlossen. Derzeit ist er Doktoratsstudent im Forschungsprojekt SmartCam am Institut für Technische Informatik an der TU Graz. Sein Forschungsschwerpunkt ist die Softwareentwicklung für intelligente, eingebettete DSP-Systeme.
Institut für Technische Informatik, TU Graz, A-8010 Graz, doblender@iti.tugraz.at

Dipl. Ing. **Arnold Maier** studierte an der TU Graz Telematik und schloss sein Studium im Juli 2002 ab. Derzeit beschäftigt er sich im Rahmen seiner Dissertation mit «Power Awareness in verteilten eingebetteten Systemen». Dieses Forschungsthema wird am Institut für Technische Informatik an der TU Graz in Kooperation mit ARC Seibersdorf research GmbH durchgeführt.
Institut für Technische Informatik, TU Graz, A-8010 Graz, maier@iti.tugraz.at

Dipl. Ing. **Roman Pflugfelder** absolvierte 2002 das Studium der Informatik im Fachgebiet Mustererkennung und Bildverarbeitung an der Technischen Universität Wien. 2001 arbeitete er als Diplomand an der University of Queensland, Australien, und erhielt für seine

Arbeit den PRIP-Studenten-Preis. Derzeit ist er bei ARC Seibersdorf research als Doktorand beschäftigt. Seine Forschungsinteressen konzentrieren sich auf den Bereich der intelligenten Videoauswertung in Videoüberwachungssystemen. Er hat einige Arbeiten und Vorträge zu diesem Thema im Bereich der Verkehrsanalyse gemacht. Sein Interesse gilt dem Technologietransfer von Videoverarbeitung in kommerzielle Produkte.
Video- und Sicherheitstechnik, ARC Seibersdorf research, A-2444 Seibersdorf, roman.pflugfelder@arcs.ac.at

Ing. **Helmut Schwabach** absolvierte 1966 die HTL für Elektronik und Nachrichtentechnik am TGM/Wien und ist seit dem Jahr 1966 im Forschungszentrum Seibersdorf, jetzt ARC Seibersdorf research GmbH, beschäftigt. Er arbeitet an der Hardwareentwicklung für eingebettete Systeme und wurde 1975 Laborleiter. Seit 1997 ist er Leiter des Geschäftsfeldes «Video- und Sicherheitstechnik», dessen Kernkompetenz die Entwicklung von Video-Surveillance-Systemen ist. Er ist wesentlich an der Entwicklung und am Design der digitalen Videosysteme für die Gebäude- und Verkehrsüberwachung beteiligt.
Video- und Sicherheitstechnik, ARC Seibersdorf research, A-2444 Seibersdorf, helmut.schwabach@arcs.ac.at

¹ Wayne Wolf ist Professor an der Princeton University, Princeton/USA; www.ee.princeton.edu/~wolf/index.html

² Jeremy R. Cooperstock arbeitet im Centre for Intelligent Machines and Department of Electrical and Computer Engineering der McGill University, Montreal/Kanada; www.cim.mcgill.ca/~jer/main.html

³ Carlo Regazzoni ist ausserordentlicher Professor am Department für Biophysik und Elektronik der Universität von Genua/Italien; <http://spt.dibe.unige.it/ISIP/Staff/rega.html>

⁴ <http://www.iti.tugraz.at/smartcam>

⁵ CIF: Common Intermediate Format. Auflösung bis 352x288 Pixel.

⁶ QCIF: Quarter Common Intermediate Format. Besitzt ¼ der Auflösung von CIF.

⁷ PAL: Phase Alternating Line. TV Standard.

⁸ fps: Frames per Second (Bilder pro Sekunde). Das menschliche Auge kann etwa 25 Bilder pro Sekunde erfassen.

⁹ MPEG: Moving Pictures Expert Group. Für die Entwicklung von Kompressionsverfahren (Audio/Video) eingesetzte Arbeitsgruppe der ISO/IEC.

¹⁰ GSM/GPRS: Global System for Mobile Communication/General Packet Radio Service.

¹¹ CMOS: Complementary Metal-Oxide Semiconductor.

¹² ADC: Analog/Digital Converter (Analog/Digital-Wandler).

¹³ GIPS: Giga-Instructions per Second. 10⁹ Instruktionen pro Sekunde.

¹⁴ FIFO: First In First Out

¹⁵ PCI: Peripheral Component Interconnect. Bussystem.

¹⁶ PTZ: Pan Tilt Zoom. Pan-Tilt-Zoom-Kamera bezeichnet Kameras, die gedreht und geschwenkt werden können sowie über ein motorisiertes Zoom verfügen.

¹⁷ Dome: Andere Bezeichnung für das Gehäuse, in dem PTZ-Kameras untergebracht sind. Auf Grund ihrer Halbkugelform werden diese Gehäuse als Dome bezeichnet.

¹⁸ I2C-Bus: Inter IC-Bus. Der Interne Integrierte Schaltungs-Bus (sprich: I-Quadrat-C-Bus) kommt mit 2 bidirektionalen Leitungen aus.

¹⁹ DSP/BIOS: skalierbarer Echtzeit-Betriebssystemkern von Texas Instruments.

²⁰ DMA: Direct Memory Access

²¹ Q9.6-Format: 16-Bit-Festkomma-Zahlenformat mit einem Vorzeichenbit, neun Vorkomma- und sechs Nachkommastellen.

²² ATEME: www.ateme.com

²³ www.national.com

Une caméra intelligente surveille la circulation

Le traitement en temps réel et le volume de données posent de hautes exigences au matériel et au logiciel

Dans le domaine des transports et de la circulation, des exigences particulièrement élevées sont posées à la sécurité et à l'exploitation des capacités. Il en résulte de nouvelles solutions de surveillance et de direction du trafic. Un excellent exemple illustrant cette tendance est donné par des caméras intelligentes de surveillance de la circulation. Outre l'enregistrement de l'information optique et la transformation de cette information en signaux électriques, les caméras intelligentes font également du traitement d'images (compression vidéo et analyse vidéo) ainsi que la communication en réseau. Les caméras intelligentes allient ainsi la prise de vues au traitement d'images et à la communication en un seul appareil embarqué.