

# Ein hochsprachenprogrammierbares System zur Vollbildauswertung im Videotakt; Anwendungen zur Interpretation monokularer, semi-strukturierter Bildfolgen bei natürlicher Beleuchtung und schnell bewegter Kamera

<sup>1</sup> Bayerische Motoren Werke Aktiengesellschaft (BMW),  
Projektgruppe ELEKTRA, Abt. EW-1, D-80788 München, FRG

<sup>2</sup> Bodenseewerk Gerätetechnik GmbH (BGT),  
Bereich FK-E, D-88641 Überlingen, FRG

<sup>3</sup> Fraunhofer Institut für Mikroelektrische Schaltungen und Systeme (IMS),  
Finkenstr. 61, D-47057 Duisburg

<sup>4</sup> Fraunhofer Institut für Zuverlässigkeit und Mikrointegration (IZM),  
Gustav-Meyer-Allee 25, D-13555 Berlin, FRG

<sup>5</sup> Institut für Neuroinformatik (INI),  
Ruhr-Universität Bochum, Geb. ND-04/5868, D-44780 Bochum, FRG

<sup>6</sup> Lehrstuhl für Numerische Mathematik und Analysis (NMA),  
Universität Passau, Innstr. 29, D-94032 Passau, FRG

**Zusammenfassung** Im vorliegenden Beitrag wird ein hochsprachenprogrammierbares System zur schritthaltenden Vollbild-Interpretation natürlich beleuchteter Szenenfolgen im Videotakt vorgestellt. Im einzelnen werden folgende Teilmodule und Subsysteme beschrieben:

- eine hochdynamische, pixellokal autoadaptive CMOS-Kamera mit ca. 120 dB Helligkeitsdynamik (20Bits/Pixel)
- ein hochsprachenprogrammierbarer *Systolic Array* Prozessor (für die pixelbezogenen Verarbeitungsmodule) im PCI-Kartenformat, samt optimierendem Compiler, Simulator und Emulator
- Systemprozeßgerüste unter Linux auf den für die Echtzeit-Anwendungen eingesetzten Hostrechnern (z.B. DEC/Alpha oder Intel/Pentium)
- eine prototypische Anwendung zur bildverarbeitungs-basierten Eigenbewegungsbeobachtung (Translationsrichtung, Rotationsraten)
- eine prototypische, automotiv Anwendung zur schritthaltenden Detektion und Kartierung des Straßen- und Spurverlaufs unter partieller monokularer 3D-Rekonstruktion, sowie
- prototypische Anwendungen zur Klassifikation verkehrsrelevanter Hindernisse (Verkehrsteilnehmer)

## 1 Einleitung und Übersicht

Im Rahmen des BMBF Förderprogramms "Elektronisches Auge" beteiligte sich die BMW AG ab Mai 1994 unter Wahrnehmung der industriellen Leitfunktion an

einem F&E-Vorhaben mit der Zielsetzung: "Schritthaltende Interpretation von Fahrscenen mit Anwendungsschwerpunkt Fahrerassistenz" [1]. Der vorliegende Beitrag gibt eine Übersicht über die erzielten Ergebnisse.

Die schritthaltende Bildfolgeninterpretation zur Fahrerassistenz stellt gegenüber anderen Anwendungen aus dem Gebiet der *Computer Vision* eine Reihe erhöhter Anforderungen an die eingesetzten Technologien und die algorithmisch-verfahrenstechnischen Grundlagen, nämlich:

- die Bewältigung szenischer Inhalte extremer bildlokaler Helligkeitsdynamik
- große Bildwiederholraten zur Lösung der Korrespondenzprobleme auch bei hoher Dynamik der Sensor-Eigenbewegung (Geschwindigkeiten bis  $250 \text{ km/h}$  und bis zu  $1 \text{ g}$  Querbeschleunigung)
- Tiefenvarianz von annähernd zwei Größenordnungen ( $5 - 200 \text{ m}$ ) bei der niedrigen Kameraeinbauhöhe von nur ca.  $1 \text{ m}$
- Einhaltung einer Gesamtlatenz der Verarbeitungskette von weniger als  $100 \text{ ms}$
- große Variabilität der szenischen Inhalte aufgrund der (durch die Straßen-Infrastruktur) nur teilstrukturierten Umfeldler
- erhebliche Zielkonflikte unter den Aspekten Rechenleistung, (Herstellungs-) Kosten, Umweltverträglichkeit (Temperatur, EMV), elektrischer Verlustleistung und Sicherheitskritikalität (Produkthaftpflicht).
- Erfordernis einer vollflächigen Bildauswertung bezüglich der sensornahen und intermediären Verarbeitungsschichten zur system- und kostenbedingten Vermeidung mechanischer Blickrichtungs- und Aufmerksamkeitssteuerung
- hohe Flexibilitätsanforderungen auch an die ikonischen Verarbeitungsprozesse; daher: Voll programmierbare Entwicklungsplattform um Hardware-Entwicklungszyklen von algorithmischen Verfahrensverbesserungen zu entkoppeln und Prozeßwechsel im Fahrbetrieb zu ermöglichen.

Aufgrund der genannten Anforderungen und Zielkonflikte waren die überwiegenden Umfänge des Projektes der Entwicklung geeigneter Technologien und Grundlagenverfahren gewidmet. Die Einzelergebnisse der verschiedenen Hard- und Software-Teilprojekte werden in den folgenden Abschnitten beschrieben.

## **2 Hochdynamischer, pixellokal autoadaptiver CMOS-Bildsensor (IMS)**

In 1997 wurde am FhG/IMS ein CMOS-Bildsensorchip samt Kamera-Hardware realisiert. Die Anforderungen an eine Kamera für den Außenraum bei verschiedensten Beleuchtungsverhältnissen sind extrem hoch. Typischerweise können Einzelbilder gleichzeitig stark unterschiedliche Helligkeitsbereiche mit reflektiertem Sonnenlicht und abgeschatteten Regionen enthalten. Die blendfreie Aufnahme von Nachtszenen bei entgegenkommenden Verkehr stellt eine weitere Anforderung dar, die bewältigt werden mußte. Um sicherzustellen, daß selbst bei solch schwierigen Beleuchtungssituationen der nachgeschalteten Bildauswertung alle wesentlichen in der Szene enthaltenen Signaturen verfügbar gemacht werden

können, mußten neue Konzepte für die Bildsensorkomponente und die Kamera-Hardware erarbeitet und implementiert werden.

## 2.1 Bildsensor-Komponente

Bei der prototypisch realisierten Bildsensorkomponente handelt es sich um einen Standard-CMOS-Chip in einer  $1\mu m$ -Technologie. Als lichtempfindliche Bauelemente wurden (wegen ihrer guten Linearität und ihrem geringen Platzbedarf) Photodioden verwendet. Die Bildmatrix besteht für diese Erstversion aus  $256 \times 256$  Pixeln. (Für Folgeversionen ist geplant, die Pixelzahl zunächst zu verdoppeln, dann zu vervierfachen.) Die einzelnen Dioden erzeugen nach einer pixellokal vorgegebenen Belichtungszeit ein helligkeitsproportionales Spannungssignal, welches von einer Ausleseschaltung am Randbereich der Bildmatrix aufbereitet und ausgelesen wird. Die Ausgangssignale aller lichtempfindlichen Elemente einer Zeile werden zeitgleich ausgewertet und können dann sukzessive ausgelesen werden. Um eine möglichst hohe Qualität der auf dem Chip gewonnenen Bildsignale zu erhalten, werden bei deren Aufbereitung im wesentlichen drei besondere Verfahren eingesetzt. Zunächst wird eine Doppelabtastung der Pixelausgangssignale vorgenommen, um Streuungen einzelner Pixel und niederfrequentes Rauschen zu unterdrücken. Weiterhin werden die Pixelausgangssignale je nach Signalwert unterschiedlich verstärkt, so daß ein möglichst großer Signal-Rausch-Abstand (SNR) der Bildsignale auch bei kleineren Signalen erreicht wird. Der erreichte maximale SNR ist größer als  $56 \text{ dB}$ . Ein limitierender Faktor für den Temperaturbereich ist der Dunkelstrom von Photodioden. Dieser ist jedoch bis zur Ecktemperatur von  $+85^\circ\text{C}$  noch so klein, daß er sich in der Bildinformation nur geringfügig bemerkbar macht. Gleichzeitig liefert der Kamerachip ein Signal, welches proportional zur Chiptemperatur ist, so daß der Einfluß der Temperatur extern kompensiert werden kann. Die untere Ecktemperatur von  $-40^\circ\text{C}$  ist unkritisch, da CMOS-Schaltungen aufgrund des geringer werdenden Dunkelstroms bei niedrigeren Temperaturen ohnehin zunehmend besser funktionieren. Drittens wird jede Diode innerhalb einer Bildperiode von  $20 \text{ ms}$  bis zu viermal unterschiedlich belichtet (siehe [3]).

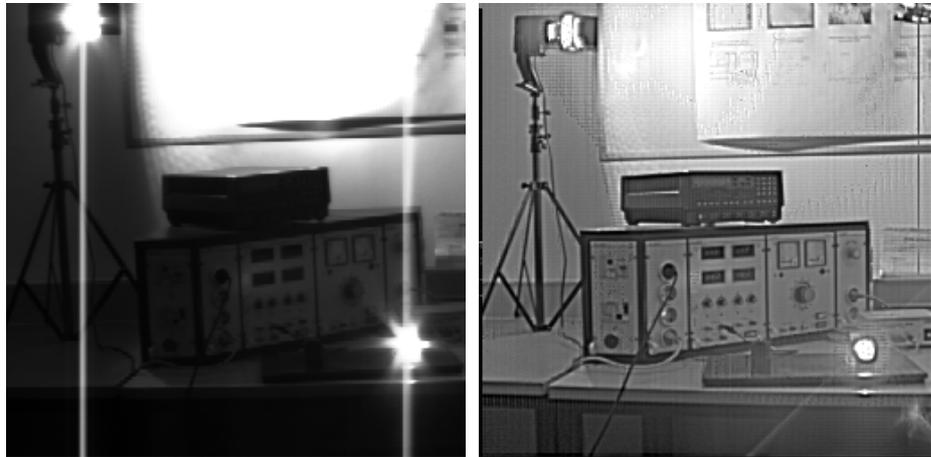
## 2.2 Kamerahardware

Die Kamera-Hardware steuert den Kamerachip dergestalt, daß jedes Pixel innerhalb einer Bildperiode von  $\frac{1}{50}$  Sekunde entsprechend der Mehrfachbelichtung bis zu viermal ausgelesen wird. Werden die im Pixel erzeugten Spannungssignale zur Kamerahardware hin ausgelesen, so muß zunächst der A/D-gewandelte Spannungswert gespeichert werden. Hinzu kommt der von der Aufbereitungsschaltung gewählte Verstärkungsbereich und die von der Ausleselogik gewählte günstigste Belichtungszeit. Die Information stückweise linearer Kennlinien für unterschiedliche Verstärkungen und Belichtungszeiten wird durch die nachgeschaltete Logik in geeigneter Weise genutzt, um den endgültigen Helligkeitswert zusammenzusetzen. Durch die Möglichkeit der unterschiedlichen Belichtung kann man die Übersteuerung der Pixel bei zu großer Helligkeit weitestgehend vermeiden und

die erreichbare Helligkeitsdynamik innerhalb eines Bildes auf nahezu 120 dB erweitern. Dies entspricht einer Anzahl von 1.000.000 darstellbaren Helligkeitswerten in einem einzigen Bild (20 *Bits/Pixel*). Im Vergleich dazu besitzen gute bis sehr gute herkömmliche CCD-Kameras eine Helligkeitsdynamik von 48 bis max. 72 dB, was etwa 256 bis 4.000 darstellbaren Helligkeitswerten entspricht (8 bis 12 *Bits/Pixel*).

### 2.3 Ergebnisse

Als Demonstrationsbeispiel wurde im Labor eine Szene mit großer Helligkeitsdynamik gestellt und mit CCD- bzw. CMOS-Sensor aufgenommen (vgl. Abb. 1). Rechts unten im Bild strahlt eine Halogenlampe direkt in das Kameraob-



**Abbildung 1.** Vergleich zwischen herkömmlicher CCD-Kamera (linkes Bild) und dem realisierten hochdynamischen CMOS-Bildsensor (rechtes Bild)

ektiv. Am linken oberen Bildrand bestrahlt ein 1000W Scheinwerfer eine Posterwand. Diese Szene wurde einerseits mit der realisierten CMOS-Kamera und zum Vergleich mit einer handelsüblichen CCD-Kamera mit automatischer globaler Belichtungszeitsteuerung aufgenommen. Das Objektiv der CCD-Kamera muß nahezu geschlossen werden, um bei direkter Einstrahlung Totalblendungen zu vermeiden. Dies verhindert dennoch nicht übersteuerte Spalten (sog. smearing-Effekte), deren Ursache "überlaufende" Ladungsakkumulate im CCD-Kamerachip sind. Aufgrund der großen Helligkeitsdynamik der Bildszene werden viele Bereiche des CCD-Kamerachips übersteuert, während andere Bereiche völlig schwarz bleiben.

Die Blende der CMOS-Kamera ist dagegen vollständig geöffnet. Da der Helligkeitsdynamikbereich der CMOS-Kamera den auf einem Qualitätsmonitor

verfügbaren Graustufenbereich von ca. 42 dB übersteigt, müssen die Informationen, die in der Bildszene enthalten sind, durch Hochpaßfilterung dem verfügbaren Graustufenbereich des Bildschirms angepaßt werden. Wie man leicht erkennen kann, enthält das Bild der CMOS-Kamera im Gegensatz zum CCD-Bild nahezu sämtliche Information der Szene.

### 3 Parallelrechnerhardware für die ikonische Bildverarbeitung (BGT)

Der Einsatz "klassischer" von-Neumann Prozessorarchitekturen ist trotz dramatischer Technologiefortschritte für anspruchsvolle, schritthaltend arbeitende Pixelverarbeitungsalgorithmen suboptimal, was zu einem schlechten Kosten/Nutzen-Verhältnis führt. Dies liegt daran, daß bei den typischen ikonischen Bildverarbeitungsprozessen in erster Linie eine große Anzahl von Operanden mit i.d.R. nur wenigen Bits Wertevorrat verrechnet werden müssen, die überdies für alle Pixel nach einheitlichen Rechenvorschriften zu bearbeiten sind. Derlei Verarbeitungsmerkmale befinden sich im Widerspruch zu der vergleichsweise geringen Operandenverfügbarkeit üblicher Hochleistungsprozessoren in Verbindung mit deren für charakteristische ikonische Verarbeitungsprozesse eigentlich zu breiten Datenpfaden von 32 bzw. 64 bit, sowie ihren Datencachestrategien, die bei typischen Bildmatrixoperationen zu unnötigen Belastungen der Hauptspeicher-Zugriffsbandbreite führen.

Die Verarbeitungsanforderungen an eine adäquate Pixelverarbeitungsmaschine legen eine Rechnerarchitektur nahe, die zum einen massiv parallele Rechenwerke nutzt und zum anderen operandengrößenspezifische Datentransfers und Rechenoperationen erlaubt ( $n$ -bit Lese-/Schreibzugriffe,  $1 \times 1$ -bit Logikoperation,  $n \times m$ -bit Addition, etc.)

Die im Projekt entwickelte Parallelrechnerhardware [2] erfüllt u.a. die oben genannten Anforderungen mit einer weiterentwickelten, hoch effizienten *SIMD*-Architektur (single instruction - multiple data). Die damit für beispielhafte ikonische Bildverarbeitungsprozesse erzielten Rechenzeiten sind je nach Anwendung um Faktoren 100 bis 10000 geringer, als bei herkömmlichen Prozessorarchitekturen (Vergleich 40MHz SIMD- und 50MHz SuperSparc-Prozessor).

Schlüsselkomponente der Rechnerstufe ist der von BGT in das Projekt eingebrachte *Systolic Array* Prozessorchip (SAP). Jeder SAP-Chip beinhaltet 64 parallel arbeitende Prozessorelemente (PE) mit zugeordneten on-chip Speicherstrukturen, die extern beliebig erweitert werden können. Zusätzlich verfügt der SAP-Chip über verschiedene, autonom arbeitende Datenpfade, die simultan zur Datenverarbeitung die Zuführung von Eingangsdaten und Entsorgung der Ausgangsdaten einschließlich Formatwandlung (bit-parallel / pixel-seriell in bit-seriell / pixel-parallel und umgekehrt) bewerkstelligen, sowie Nachbarschaftsoperationen in einer  $\pm 4$ -Umgebung ohne zusätzliche Zyklen unterstützen.

Vier dieser SAP-Chips sind zusammen mit externer Speichererweiterung auf einem Aufsteckmodul (SAP-Slice) kaskadiert und realisieren eine parallel arbeitende Zeile von 256 Prozessorelementen (PE). Bis zu zwei SAP-Slices können

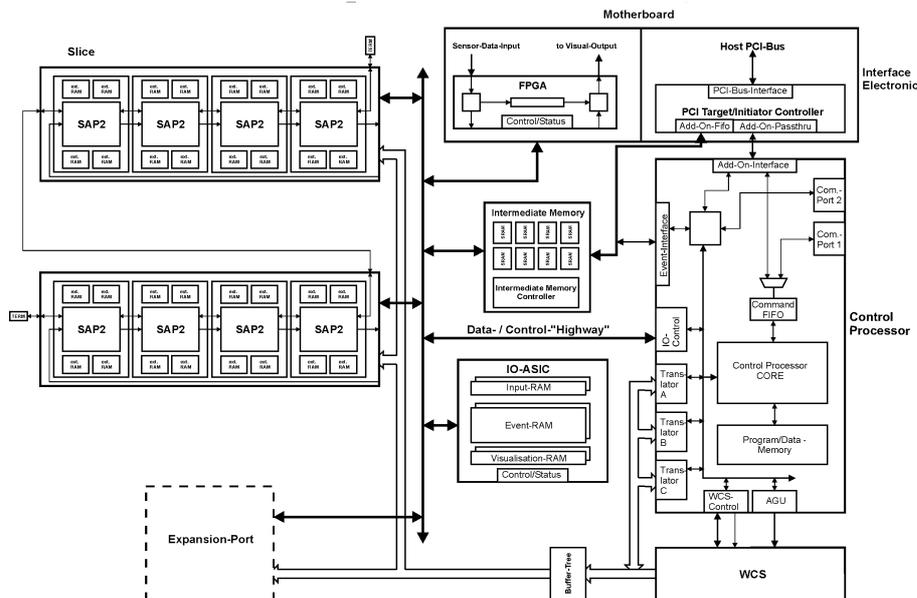


Abbildung 2. Blockschaltbild der Systolic Array Baugruppe

auf der SAP-Trägerplatine (belegt einen PCI-Slot) aufgesteckt werden und erlauben damit Konfigurationen von bis zu 512 PEs. Eine rückseitige Bestückung von zwei weiteren SAP-Slices ist vorgesehen (Konfigurationserweiterung auf 1024 PEs). Die zeilenförmige Aneinanderschaltung von PEs wird durch ein ausgefeiltes Speichermanagement ergänzt, das dem Anwender in Verbindung mit dem mächtigen Compiler die programmierte 2D-Verarbeitung von Bilddaten auf einfachste Art und Weise ermöglicht. Darüberhinaus erlaubt die statische und dynamische Speicherverwaltung bildübergreifende Operationen, sowie das Arbeiten mit Referenzbildern. Diese vielfältigen Funktionen zur Speicherverwaltung sind zusammen mit einer hoch flexiblen Ablaufsteuerung der SAP-Programme und einer dynamischen Parameterverwaltung auf einem einzigen Control-ASIC implementiert. Neben dem Control-ASIC umfaßt die SAP-Trägerplatine weitere Funktionsblöcke, die die Einbindung der Rechnerstufe in handelsübliche Workstations der neuesten Generation unter Berücksichtigung der jeweils projektspezifischen Anwendungsanforderungen erlauben.

Die 14-lagige SAP-Trägerplatine (vgl. Abb. 3) enthält außer dem Control-ASIC folgende weitere Funktionsblöcke:

- eine adaptierbare digitale Sensorschnittstelle
- das I/O-ASIC zur programm- und konfigurationsabhängigen Aufbereitung der SAP-Eingangs- und -Ausgangsdatenströme
- den Instruktionsspeicher für den SAP-Binärcode (WCS)
- 4 MB intermediären Speicher zur flexiblen, anwendungsspezifisch formatierten Bereitstellung der SAP-Ausgangsdaten

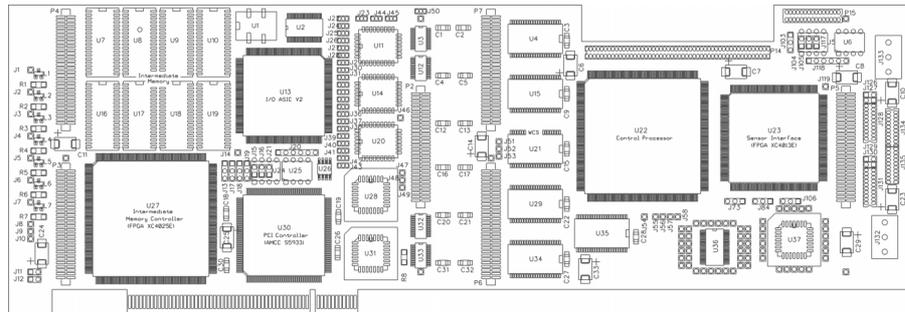


Abbildung 3. CAD-Plot der SAP-Trägerplatine

– das PCI-Bus Interface

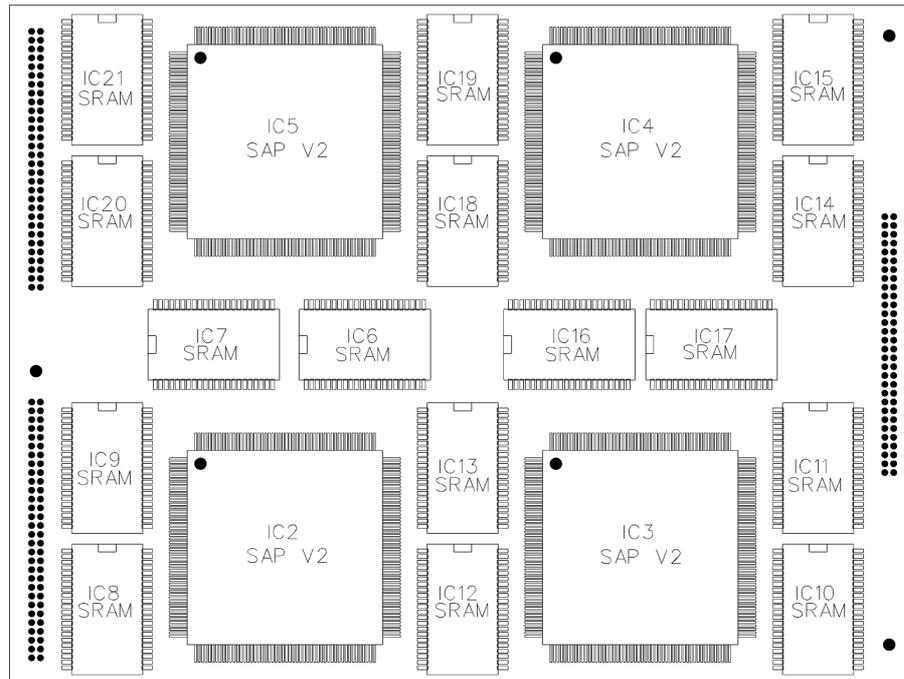
Ein Blockschaltbild der SAP-Platine ist in Abb. 2 gezeigt, ein CAD-Plot der Trägerplatine und zweier SAP-Slice-Varianten findet sich in Abb. 3, 4 und 5. Die in Abb. 5 gezeigte Variante des SAP-Slices verwendet anstelle der lediglich auf Komponentenbasis miniaturisierten ASICs und SRAMs, sog. MCMs (vgl. Abschnitt 3.2).

Bauvolumen	450cm <sup>3</sup>
Verlustleistung	20 W
I/O Rate	160/160 MByte/sec
Maximale Speicherzugriffsbandbreite	10 GByte/sec
Nachbarschaftszugriffe ohne zusätzliche Zyklen	[−4, +4]
Rechenleistung (8Bit ADD/SUB)	2, 2 GOPS
Rechenleistung (Multiply/Accumulate)	0, 4 GOPS

Tabelle1. Leistungsmerkmale der SAP/PCI Platine mit 512 Prozessor-Elementen

### 3.1 Einsatz der SAP-Rechnerstufe als PCI-Einsteckkarte

Die entwickelte Parallelrechnerstufe wird derzeit in einer Konfiguration mit 512 PEs zur zeilenweise parallelen Verarbeitung von Bilddaten eingesetzt. Die sequentiell angelieferten Pixeldaten werden in einen spaltenparallel anfallenden Datenstrom gewandelt und nach Verrechnung einer geeigneten Anzahl von Zeilen (oder auch Bildern) spaltenparallel ausgegeben, formatgewandelt und im intermediären Speicher ablegt. Der Transfer der Ergebnisdaten in den Hauptspeicher des Wirtsrechners erfolgt im *PCI-Master-Write-Modus* und belastet daher allenfalls die Speicherbandbreite (bei *single-port-memory*), nicht aber die CPU(s)

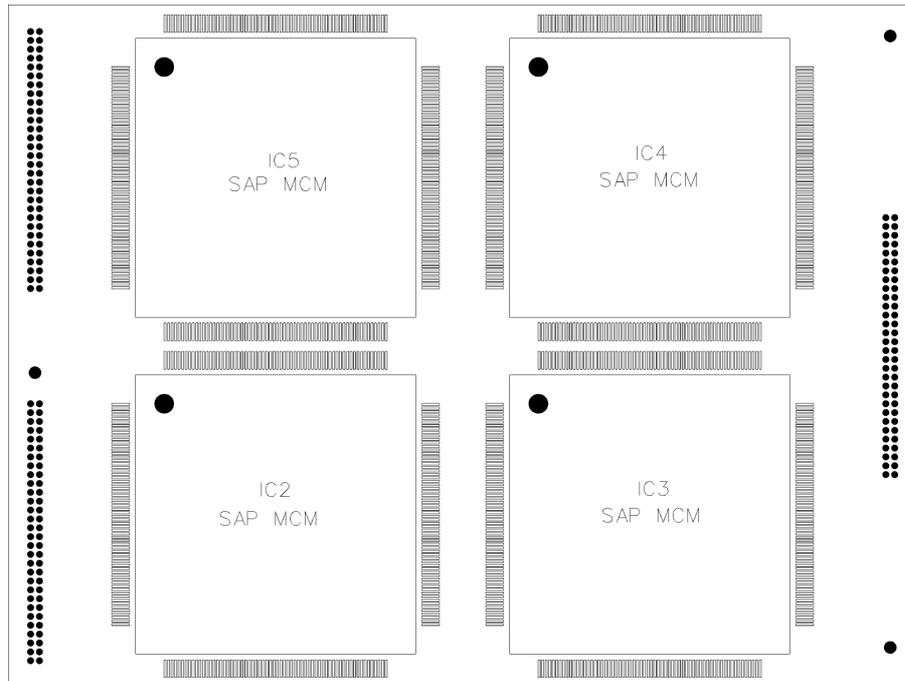


**Abbildung 4.** SAP-Slice auf Komponentenbasis

des Wirtsrechners. Die mit der entwickelten Baugruppe gemessene PCI-Master-Write-Bandbreite von  $128 \text{ MB/s}$  realisiert praktisch den theoretischen Spitzenwert ( $133 \text{ MB/s}$ ). Tabelle 1 faßt einige der Leistungsmerkmale einer mit 512 Prozessorelementen bestückten SAP-Platine zusammen.

### 3.2 Modularer Aufbau planarer Multi-Chip-Module (IZM)

Langfristige Zielrichtung dieses Teilprojekts war die über die Entwicklung hochintegrierter ASICs hinausgehende Miniaturisierung mit Methoden der Aufbau- und Verbindungstechnik durch 3-dimensionale Integration (Komponenten-Stapelung). Dem diente ein Konzept zur planaren Hybridintegration in Multi-Chip-Modulen (MCM) [24]. In keramische 4"-Substrate (AIN) werden per YAG-Laser präzise Fensteröffnungen zur Aufnahme ungehäuster IC-Komponenten geschnitten. Eine transparente, laminierte Adhäsionsfolie auf der Substratfrontseite bildet die Referenzfläche für alle einzubettenden Komponenten. Unter der Optik eines Bestückungsaligners werden die Komponenten nacheinander in den Öffnungen mit Hilfe einer Referenzmaske justiert und an der Adhäsionsfolie temporär fixiert. Durch eine Epoxy-Fügung - von der Rückseite in die umlaufenden Fugen eingebracht - werden die Komponenten in ihrer Lage dauerhaft fixiert. Nach Abzug der Adhäsionsfolie liegt wieder eine geschlossene planare



**Abbildung 5.** SAP-Slice im MCM-Aufbau

Substratoberfläche vor, auf der in starrer Maskentechnik eine Dünnschichttechnik-Mehrschichtenverdrahtung aufgebaut werden kann, die die Komponenten untereinander verbindet und Bondpads bzw. Lötperlen in oberster Lage zur Boardkontaktierung des Moduls bietet. Ausschließlich Dünnschichtprozesse werden angewandt, um die IC-Ankontaktierung und die mehrschichtige Cu-Verdrahtung im Polyimid-Dielektrikum zu erstellen. Die Ankontaktierung erfordert keine spezielle Vorbehandlung. ICs unterschiedlicher Technologie können miteinander integriert werden. Die Einbettung kapazitiver Bauteile und optischer, elektrischer, thermischer Durchführungen ist ebenso möglich. Planare Einbettmodule realisieren bei dichter Komponentenfüllung einen kompakten MCM-Aufbau, dessen Höhe die Substratdicke kaum überschreitet. Diese Einbettmodule, zusätzlich versehen mit Substratdurchkontaktierungen, erlauben durch die Mehrschichtenverdrahtung hindurch die Realisierung kurzer, vertikaler Signalwege. Ergänzt durch flächenhaft angeordnete Lötverbindungen auf der Ober- und Unterseite der Module, kann die Stapelung von MCMs mit extrem kurzen, vertikalen Busleitungen erreicht werden.

Im Projektzeitraum konzentrierte man sich zunächst auf den Bau von planaren Einbettmodulen, die zwar noch nicht stapelfähig sind, aber als modulare Einheit (Prozessor, Speicher, Abblock-Kondensatoren) in die vom BGT entwickelte Systemarchitektur eines SIMD-Rechners passen. Dadurch lassen sich

die gehäusten Einzelkomponenten auf den SAP-Slices ersetzen. Das SAP MCM der Größe  $32 \times 32 \text{mm}^2$  enthält 4 SRAMs ( $64K \times 16$ ) und 4 Puffer, die eine SAP Komponente (64 PEs) umgeben. Die erstellten Einbettmodule werden funktionsgeprüft und per Drahtbond in ein Standardgehäuse gebaut. Mit den so gewonnenen SAP MCMs können die SAP-Slices alternativ bestückt werden (ein CAD-Plot eines MCM-Slices ist in Abb. 5 gezeigt).

### 3.3 Software-Entwicklungswerkzeuge für die SAP-Baugruppe

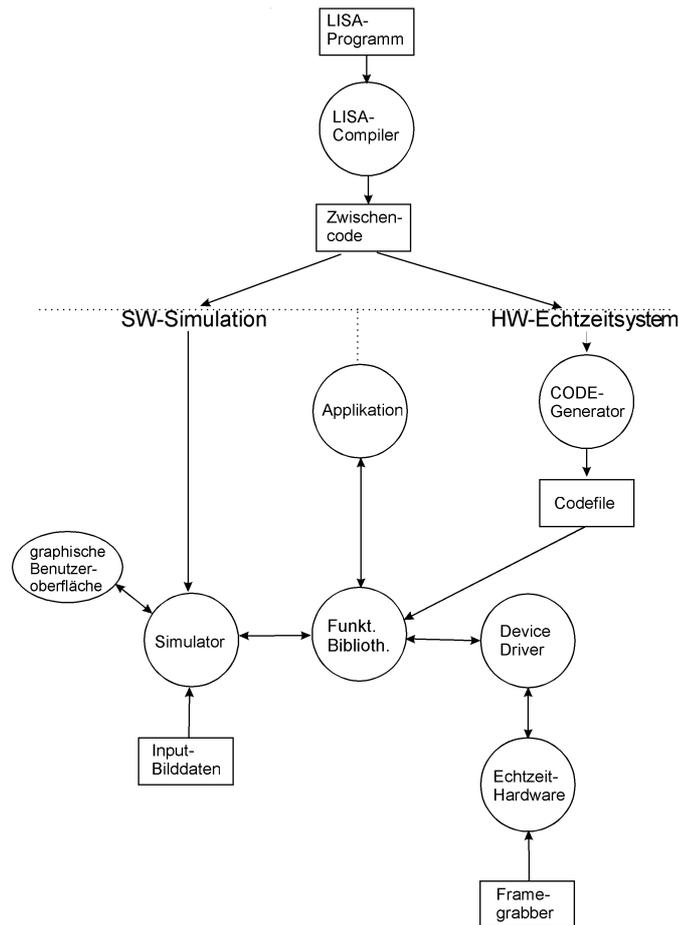
Parallel zur Entwicklung des SIMD-Prozessors wurde eine zu seiner Programmierung geeignete Hochsprache definiert und die zu ihrer Nutzung erforderlichen Software-Entwicklungswerkzeuge realisiert. Die SAP-Entwicklungsumgebung umfaßt u.a. folgende Instrumente und Werkzeuge:

- die C-ähnliche SIMD-Programmiersprache *LISA* (*L*anguage for *I*mage processing on *S*ystolic *A*rrays), die zum einen den arithmetisch-logischen Instruktionssatz von C, darüber hinaus aber auch Struktur- und Bit-Feld-artige Konstrukte enthält. Den Anwendungen entsprechend werden bildlokale Operationen wie zweidimensionale, bildübergreifende Operationen wie dreidimensionale Arrays indiziert. Die "Schleifenverwaltung" ist nicht Gegenstand des Quellencodes, sondern wird vom Compiler generiert.
- einen optimierenden *LISA*-Compiler, der nach dem Übersetzungsvorgang die Anzahl der zur Laufzeit erforderlichen Prozessorzyklen, sowie die Größe des bei der Programmausführung benötigten Speicherplatzes ausweist (*Profiler*-Funktion).
- den SAP-Simulator (lauffähig unter Solaris, SunOS, Linux, AIX, HP-UX), welcher die Berechnungsvorgänge der SAP-Hardware nachbildet, die Visualisierung beliebiger Zwischenergebnisse erlaubt und es gestattet, die SAP-Ergebnisdaten in beliebige, auf dem SAP-Hostrechner ablaufende, symbolische Verarbeitungsmodule einzuspeisen.
- den SAP-Emulator, der die SAP-hardwareunterstützte Verarbeitung auf Festplatte gespeicherter, digitaler Bildfolgen ermöglicht, wobei dann u.a. auch ein Bildeinzelschrittmodus zur Verfügung steht, mit dem die Visualisierung von spezifischen ikonischen Zwischenergebnissen möglich ist.

Die Steuerung von SAP-Anwenderprogrammen aus sensorfernen, symbolischen Verarbeitungsprozessen erfolgt über eine *Linux*-Anwenderbibliothek, welche die erforderliche Gerätetreiberfunktionalität zur Verfügung stellt. Eine Übersicht über die verschiedenen SAP-Softwarewerkzeuge und ihr Zusammenwirken ist in Abb. 6 gegeben.

## 4 Kameramodellierung und Kamerakalibrierung (NMA)

Zur Erhöhung der Vermessungsgenauigkeit szenischer Inhalte im Fernbereich und zur Nutzung der bei den ikonischen Prozessen berechneten Subpixelgenauigkeiten ist es von entscheidender Bedeutung, möglichst genaue Abbildungsvorschriften zwischen Bildmatrix und Szene zur Verfügung zu haben. Die in diesem Abschnitt beschriebenen Ansätze dienen diesem Ziel.



**Abbildung 6.** SAP-Entwicklungswerkzeuge

Generelle Aufgabe der Kameramodellierung und Kamerakalibrierung ist die approximative Beschreibung und Ermittlung der Abbildungseigenschaften zwischen Punkten eines gegebenen 3D-Koordinatensystems und den Pixelpositionen der in die Sensorebene abgebildeten Objekte. Die Adäquatheit des gewählten Kameramodells, sowie die numerische Genauigkeit der dafür ermittelten Modellparameter ist entscheidend für die Güte der Tiefen- und Bewegungsschätzung aus monokularen oder polyokularen Bildfolgen.

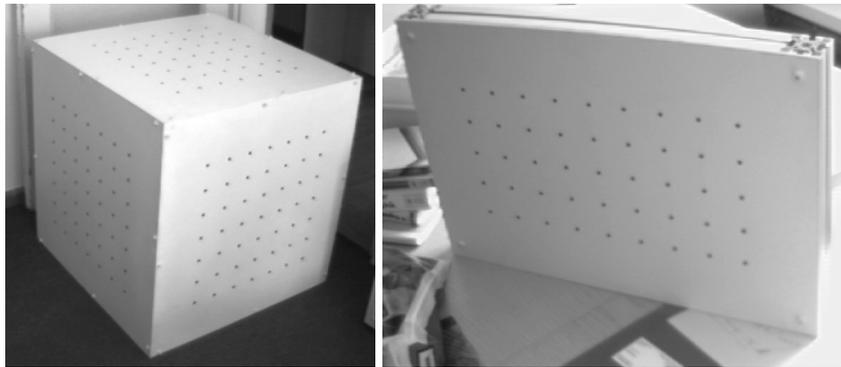
Zur Bildsensorkalibrierung zählen neben der Bestimmung der Kameraparameter aus Bildern von Eichkörpern bekannter Geometrie auch Algorithmen zur subpixel-genauen Detektion projizierter 3D-Marken im Kamerabild und zur automatischen Zuordnung von 2D-Bildpositionen zu den vermessenen 3D-Ortskoordinaten.

Eine zusätzliche Verbesserung der numerischen Stabilität der resultierenden Optimierungsprobleme, sowie der daraus zu berechnenden Kameraparameter, wird durch die Erweiterung der üblichen Einbildkalibrierung auf ein semi-automatisches Mehrbildverfahren erreicht.

Die im Projekt entwickelte Modellierung erlaubt über die perspektivische Projektion hinaus auch die Einbeziehung von Korrekturen für folgende typische Verzeichnungsarten realer optischer Systeme:

- Radiale Verzeichnung durch die Krümmung der Linsenoberflächen;
- *“Decentering Distortion”* durch nicht exakt lineare Anordnung der verwendeten Einzellinsen;
- *“Thin Prism Distortion”* durch Verkippung einzelner Linsenelemente bzgl. der optischen Achse und/oder der Bildsensormatrix.

Für eine prototypische Kalibrieranordnung wurden als 3D-“Eichkörper” ein Würfel mit identischen Seitenflächen, sowie (wahlweise zur Ermittlung der Kameraposition im Fahrzeug) eine fahrzeugstationäre Platte gewählt, auf denen jeweils äquidistante Markierungen aufgetragen sind. Abb. 7 zeigt Bilder der verwendeten Kalibrierkörper. Ein Vergleich der erzielten Ergebnisse der Kalibrie-



**Abbildung 7.** Zur Kamerakalibrierung verwendete 3D-“Eichkörper”

rung mit und ohne Berücksichtigung der o.a. Verzeichnungen zeigt, daß letztere bei der Projektion gegebener 3D-Punkte zu Abweichungen der zugehörigen Bildposition in der Größenordnung von mehreren Pixeln (am Bildrand bis zu 15 Pixeln) führen können. Umgekehrt treten bei der Rekonstruktion der 3D-Information aus Bilddaten (beispielsweise bei der Abstandsschätzung von Objekten) allein durch die nicht ausreichend genaue Modellierung der Kameraabbildungseigenschaften relative Fehler von mehr als 10 Prozent auf.

Für die möglichst exakte Eigenbewegungsbestimmung des Fahrzeugs, die 3D-Rekonstruktion des Straßenverlaufs im Fernbereich und eine verbesserte Ab-

standsschätzung von Hindernissen ist daher die Einbeziehung von optischen Zeichnungen bei der Kameramodellierung unerlässlich.

## 5 System-Architektur (BMW)

In Anbetracht des rapiden technologischen Fortschritts im Bereich der Standard-Hochleistungsprozessoren war es von grundsätzlicher Bedeutung, bei der Wahl des Hostrechners auf technisch führende, voll im "Mainstream" des Marktes liegende Systeme zu setzen.

Dabei spielten Verfügbarkeits- und Kostengesichtspunkte ebenso eine Rolle, wie die unbedingte Kontinuität der erzielten Projektergebnisse, d.h. deren Abkopplung von schwer vorhersehbaren Marktentwicklungen im Bereich der Halbleiter- und Rechnerindustrie.

Entsprechend zog dies die Entscheidung nach sich, ein nichtproprietäres Betriebssystem zu verwenden. Nach diversen Evaluierungen vereinbarten die Industriepartner im Projektverbund auf *Linux* und *GNU*-Entwicklungswerkzeuge (inkl. *X11/XFree*) zu setzen.

Als Hardware-Plattformen wurden zunächst DEC/Alpha und Intel/Pentium basierte Systeme gewählt.

Die genannten Festlegungen zogen nach sich:

- die Erfordernis der Implementierung *Linux*-konformer Gerätetreiber für die SAP- und Framegrabber-Baugruppe und die
- die anwendungsgerechte Programmierung der erforderlichen (aufgrund der noch fehlenden POSIX-Erweiterungen nur eingeschränkt echtzeitfähigen) *Linux*-Systemdienste für die Echtzeitapplikationen.

Insbesondere ging es darum, den strikt synchron anfallenden, die gesamte symbolische Prozeßkette synchronisierenden Ergebnis-Output des *Systolic Array* Prozessors unter Verwendung von applikationsunabhängigen Systemprozessen (Visualisierung und Fahrzeug-Meßdatenerfassung, etc.) von den symbolischen Anwendungsprozessen zu entkoppeln. Die letztgenannte Erfordernis beruht u.a. auf dem Faktum, daß die späten (d.h. sensorfernen) Bildinterpretationsprozesse aufgrund ihrer erhöhten Komplexitätsmaße nur im statistischen Mittel einiger weniger Frames synchron gehalten werden können. Ergebnis entsprechender Untersuchungen ist, daß das zeitliche Antwortverhalten derzeit verfügbarer *Linux*-Distributionen für Bildwiederholraten von 50 bzw. 60 Hz auch unter Verzicht auf die genannten POSIX-Erweiterungen ausreichend ist, und die erwähnte, zeitliche Entkopplung für ganze Klassen von Anwendungen einheitlich realisiert werden konnte. Das in Abb. 8 gezeigte Blockdiagramm der hierfür implementierten Systemprozesse dient nicht nur als Grundgerüst für die in Abb. 9 dargestellte Gesamtapplikation zur Bildfolgeninterpretation, sondern bildet auch die Basis für die im Projekt entwickelte Software zur Bild- und Meßdatenerfassung. Im letztgenannten Falle besteht die "Applikation" lediglich aus der Funktion, den anfallenden Bild- und Meßdatenstrom von bis zu 15 MB/s durch das *Linux*-Dateisystem schritthaltend auf Festplatte zu speichern. Die dabei durch betriebssystembedingte File-Cache-Strategien auftretenden Fluktuationen der effektiven

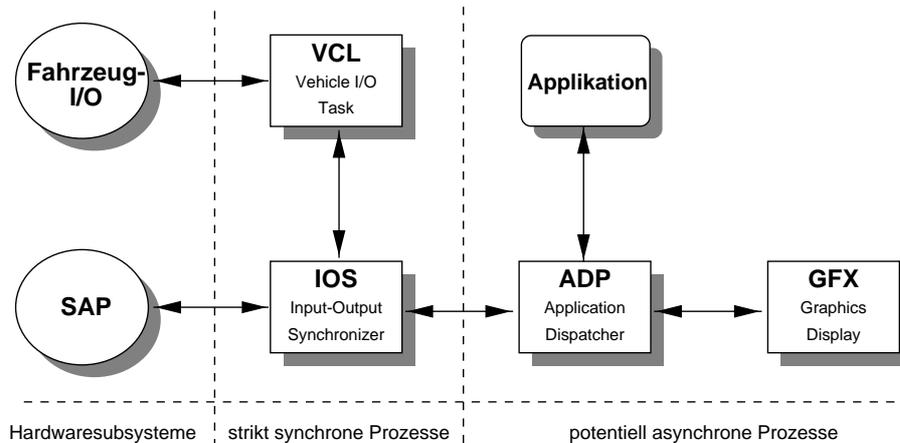


Abbildung 8. Systemprozesse

Schreibdatenrate ziehen transiente Latenzen von bis zu  $120ms$  nach sich, die zum einen im statistischen Mittel ohne Datenverluste wieder ausgeglichen werden können und zum anderen aber den vom Frame-Grabber per PCI-Master-Write in den Hauptspeicher kopierten, synchronen Datenstrom in keiner Weise beeinträchtigen.

## 6 Ikonische Prozesse auf dem Systolic Array Prozessor (BMW/BGT)

Zunächst wird ein Programm zur Berechnung des kantenbasierten optischen Flusses vorgestellt, bei dem die Kantengeschwindigkeits-Normalkomponenten für beliebig auftretende Konturpixel bildflächendeckend im Videotakt bestimmt werden. Dazu erforderlich sind:

- ein Modul zur Berechnung von Binärkanten samt morphologischer Bereinigung, Skelettierung und Attributierung
- ein Modul zur attributorientierten, bildübergreifend eindeutigen Konturpunktzuordnung, inklusive der Bestimmung der aktuellen Positionsverschiebung senkrecht zur Kante, sowie
- eine pro Bild für alle Konturpixel durchgeführte Kalmanfilterung der rohen Verschiebungsdaten zur Gewinnung der dynamisch-optimalen Flußgeschwindigkeitskomponenten.

Zur Charakterisierung der Leistungsfähigkeit des entwickelten Parallelrechners sei erwähnt, daß die beschriebenen Operationen gemäß den Synchronitätseigenschaften des SIMD-Prozessors auf einer einzigen SAP/PCI-Platine in Abhängigkeit von der Bildwiederholrate von  $60Hz$  ca. 8 Millionen mal pro Sekunde durchgeführt werden ( $60 \times 512 \times 256 \frac{Pixel}{Bild \times Sekunde}$ ).

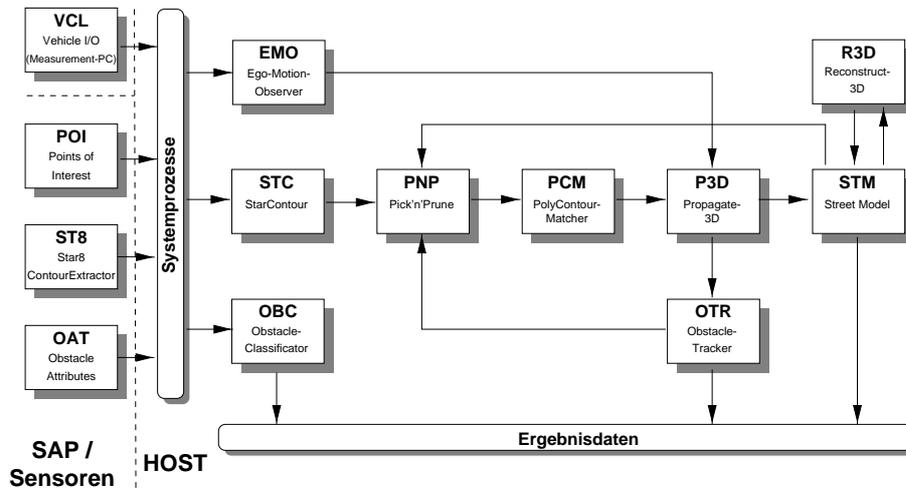


Abbildung 9. Applikationsprozesse "Elektronisches Auge"

In einer anderen in Abb. 9 dargestellten Gesamtapplikation wird die außerordentliche Ausgabebandbreite (128 *bit/Pixel*) des SAP-Prozessors genutzt, um die vorsortierten Ergebnisse verschiedener ikonischer Verarbeitungsprozesse im Bildtakt simultan bereitzustellen. Im einzelnen werden berechnet:

- die zum Aufbau eines Konturnetzwerks erforderliche richtungscodierte Kanten-"Ereignis"-Information (ST8),
- die pixelbasierten Hindernisattribute zur Objektklassifikation (OAT) und
- die "point-of-interest"-Bestimmung und Attributierung zur Realisierung eines Bildsignal-basierten Eigenbewegungsbeobachters (POI) (vgl. Abschnitt 7);
- weiterhin wird zu Zwecken der Visualisierung das Original-Grauwertbild ausgegeben.

Bei der erwähnten Extraktion der Konturskelett-Information handelt es sich um die Bestimmung von pixellokal ortstreuen Kantenpositionen und Topologien. Die Ortstreue beeinträchtigende Tiefpaßfilterungen werden dabei vermieden. Unter dem Aspekt der Bildkomponentenmarkierung basiert das in  $\alpha$ -Version realisierte Verfahren nicht auf einer binären, sondern einer polyvalenten Bildkomponentenzerlegung. Abb. 10 zeigt die mittels SAP-Baugruppe aus dem dargestellten Graubild extrahierten Konturen (Anm.: das zugrunde liegende Bildmaterial wurde noch im suboptimalen Videoformat "field integration / interlace" aufgezeichnet).

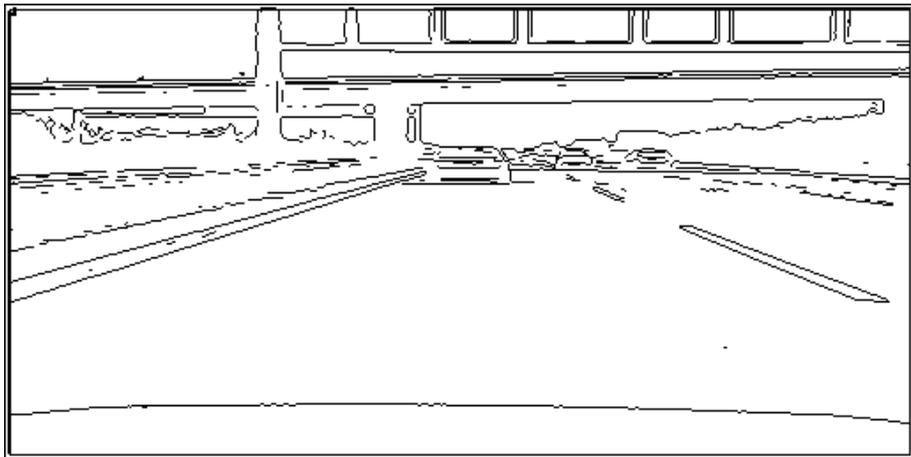


Abbildung 10. Konturextraktion

Der auf SAP implementierte Algorithmus erfordert ca. 50 9-bit-Operationen und 700 1-bit-Operationen pro Pixel, die sich pro Bildpunkt zu ca. 1150 1-bit-Operationen addieren. Bei einer Bildwiederholrate von 50Hz und 256 Bildzeilen führt dies zu einem erforderlichen Rechendurchsatz von knapp 15 Millionen 1-bit-Instruktionen pro SAP-Prozessorelement, wobei allerdings insgesamt 512 PEs zur Verfügung stehen. Im Gegensatz dazu benötigt eine konventionelle CPU ca. 750 (32- oder 64-bit-) Instruktionen pro Bildpunkt bei  $512 \times 256$  Pixeln und 50Hz Bildwiederholrate, also ca. 5 Gigainstruktionen. Der "Overhead" für 30-40 Doppelschleifensteuerungen ist hierbei noch nicht berücksichtigt.

Der mittels des ST8-Moduls gegebene Vergleich der auf SIMD-Prozessor bzw. konventioneller CPU erforderlichen Anzahl von Instruktionen zeigt klar die Vorteile der (mit der Bildbreite skalierbaren und der Bildhöhe kaskadierbaren) SAP-Parallelhardware anhand eines konkreten Anwendungsbeispiels.

Das eingangs angeführte Quadruplett von Pixelverarbeitungsmodulen dient der symbolischen Verarbeitungskette als Basis zur simultanen Kartierung des Fahrbahn- und Spurverlaufs, zur Berechnung der Kameraeigenbewegung, zur Hindernisklassifikation und überdies zur Visualisierung von Ergebnisdaten.

## **7 Bestimmung des fahrzeugdynamischen Zustands; Eigenbewegungsbeobachter (NMA/BMW)**

Eine möglichst zeitnahe Bestimmung der Fahrzeug-Rotationen (Gieren, Nicken, Wanken) und der Fahrzeugbewegungsrichtung (Schwimmwinkel) verringert die Komplexität der nachfolgenden symbolischen Algorithmen zur 3D-Rekonstruktion und Fahrbahnkartierung und verkürzt damit deren Laufzeit wesentlich.

Zur Berechnung der Orientierungskinetik hat sich die Verwendung von zeitlich stabilisierten Punktverschiebungen als nützlich erwiesen. Dabei werden mit Hilfe der vom ABS-System gelieferten, skalaren Geschwindigkeit die vektoriell translatorischen und rotatorischen Bewegungsanteile des Fahrzeugs aus Paaren aufeinander folgender Bilder bestimmt.

### **7.1 Beschreibung der zugehörigen Teilmodule**

**Extraktion von markanten Punkten aus Bildsequenzen (POI).** Für die Detektion markanter Punkte werden Kriterien, basierend auf der expliziten Berechnung von Richtungskrümmungen (partielle zweite Ableitungen des diskreten Grauwertverlaufes) oder aber Eigenschaften des Krümmungstensors verwendet. Für letztere ist das aus der Literatur bekannte Verfahren von Harris und Stephens [4] ein typischer Vertreter.

**Attributierung der extrahierten markanten Punkte (POI/EMO).** Um eine Zuordnung der pro Halbbild unabhängig voneinander durch die SAP-Parallelhardware extrahierten Punkte über die Zeit zu gewährleisten, werden die

extrahierten markanten Punkte mit Attributen versehen, die ihre lokale Bildumgebung beschreiben. Als Merkmale dienen dabei die Koeffizienten einer lokalen Passung des Bildinhalts mit orthogonalen Polynomen niedrigen Grades.

**Bildübergreifende Zuordnung mit Subpixelgenauigkeit (EMO).** In diesem Teilmodul wird eine Zuordnung von Punkten aufeinander folgender Frames mit Hilfe der bereitgestellten Attribute und unter Verwendung vorher ermittelter Information (d.h. bereits einige Male wiederaufgefundener Punkte) durchgeführt. Eine Verfeinerung der Verschiebungsinformation auf Subpixelgenauigkeit wird erreicht, indem man diejenige Verschiebung berechnet, bei der die Abweichung zwischen den entsprechend translierten Polynomapproximationen minimal wird.

**Berechnung der Fahrzeugeigendynamik aus den ermittelten Punktverschiebungen (EMO).** Durch eine neuartige, halb-perspektivische Modellierung der bei der Bestimmung der Kameraeigenbewegung zu minimierenden Fehlerfunktion, konnte eine entscheidende Verbesserung der Ergebnisse im Vergleich zu den aus der Literatur bekannten Ansätzen und Verfahren [5–12] erzielt werden. Die verwendete halb-perspektivische Modellierung, die nur Rekonstruktionen von Punkten *vor* der Kamera erlaubt, ist für eine robuste und ausreichend exakte Bestimmung der Kameraeigenbewegung vor allem im Fall von selbstbewegten 3D-Objekten in der betrachteten Szene unabdingbar. Eine ausführliche Darstellung der entwickelten (u.a. semi-perspektivischen) Methoden findet man in [13]; ein weiterer Artikel ist zur Veröffentlichung [14] eingereicht.

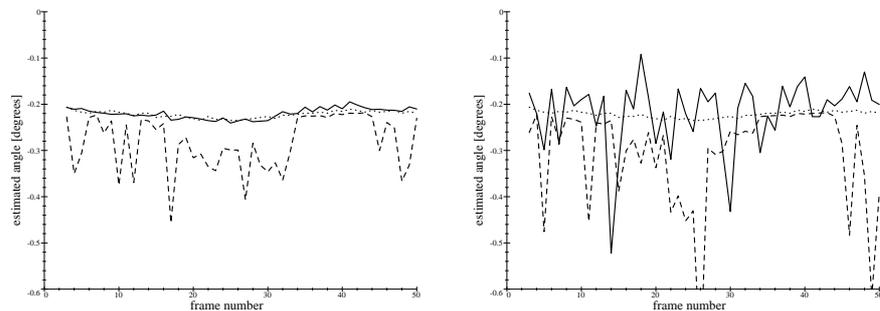
## 7.2 Ergebnisse

Abb. 11 zeigt einen Vergleich der Ergebnisse des oben beschriebenen Verfahrens für 50 Bilder einer Sequenz am Beispiel des Gierwinkels  $\phi_x$ , sowie die zugehörigen Daten des Gyrosensors (Punkte). Die gestrichelten Kurven stellen die berechneten Ergebnisse bezogen auf alle extrahierten und erfolgreich zugeordneten markanten Punkte dar, basierend auf der Minimierung der hier entwickelten Zielfunktion  $F_0$  (semi-perspektivischer Ansatz) und einem nichtlinearen Standardverfahren (mit zugehöriger Zielfunktion  $F_W$ ) von Weng, Huang und Ahuja [7]. Die durchgezogenen Kurven repräsentieren die Ergebnisse nach fünf Iterationen unter Berücksichtigung einer Verfeinerung, bei der Ausreißer (z.B. Punkte, die auf eigenbewegten Objekten liegen) eliminiert werden.

Bei dem hier entwickelten Ansatz ist eine deutliche Verbesserung der Resultate in Bezug auf Genauigkeit und Stabilität erkennbar. Im Gegensatz dazu werden die mit Hilfe der Zielfunktion  $F_W$  berechneten Ergebnisse durch die Elimination von Ausreißern nur unwesentlich verbessert.

Die Genauigkeit der hier durch Bildverarbeitung bestimmten Ergebnisse ist denen handelsüblicher Gyrosensoren durchaus ebenbürtig, in manchen Fällen

sogar überlegen, da die entsprechenden Informationen unmittelbar aus dem optischen System gewonnen werden und so Übertragungseffekte (z.B. durch dynamische Filtertechniken) und insbesondere die bei Gyrosensoren auftretenden Temperatureffekte vermieden werden.



**Abbildung 11.** Originalbild (oben) und berechnete Ergebnisse der Gierwinkelraten  $\Delta\phi_x$  basierend auf den Zielfunktionen  $F_0$  (unten links) und  $F_W$  (unten rechts); Ergebnisse ohne iterative Verbesserung (gestrichelt), Ergebnisse nach Elimination von Ausreißern (durchgezogen) und entsprechende Daten des Gyrosensors (gepunktet).

## 8 Straßen- und Hindernismodellierung (BMW/NMA)

Die Straßen- und Hindernismodellierung besteht aus einer Prozeßkette, die sich aus Modulen zur Extraktion von Konturen aus Bilddaten, deren Vorauswahl in interessierenden Bereichen, der Polygonzugpassung zur Datenreduktion, der Zuordnung von Konturteilen zur Gewinnung eines zeitlichen Verlaufs, bis hin zur Rekonstruktion der dreidimensionalen Positionen der Fahrspuren und Hindernisse im ortsfesten 3D-Koordinatensystem zusammensetzt.

**Komponentenmarkierung (STC).** Das bei BMW entwickelte Modul zur Komponentenmarkierung nutzt die vom SAP-Prozeß ST8 generierten

Kantenskelett-Ereigniscodes, um daraus eine symbolische Repräsentation der Bildregionen samt Attributen und die sie begrenzenden Konturen aufzubauen. Die Konturknoten (End- und Verzweigungspunkte) sind in dieser Darstellung mittels zyklisch und sequentiell vierfach verketteter Listen repräsentiert, die dazwischen liegenden Polygonzüge (8 Freeman-Richtungen) sind lauffängencodiert, um die Datencache-Strategien moderner Workstationprozessoren effizient zu nutzen. Auf Wunsch kann neben der Netzwerktopologie der Randkonturen auch die Regionen-Einschlußhierarchie ausgegeben werden.

**Konturselektion (PNP).** Bei der Verarbeitung der aus den Vollbildern berechneten reichhaltigen Repräsentation aller Konturskelette stoßen die Algorithmen zur Straßen- und Hindernismodellierung an die ihnen durch die Echtzeitbeschränkung der Gesamtanwendung gesetzte Grenze. Um dieser Einschränkung genügen zu können, ohne auf die Details der Kanteninformation verzichten zu müssen, selektiert ein spezielles Modul der Prozeßkette auf Anforderung der die Kanteninformation nutzenden Prozesse die gewünschten Konturuntermengen anhand vorgegebener geometrischer Muster (z.B. Trapezfolgen). Weitere Attribute einer Anforderung sind u.a. Kantenrichtung, Länge, Gradient, etc.

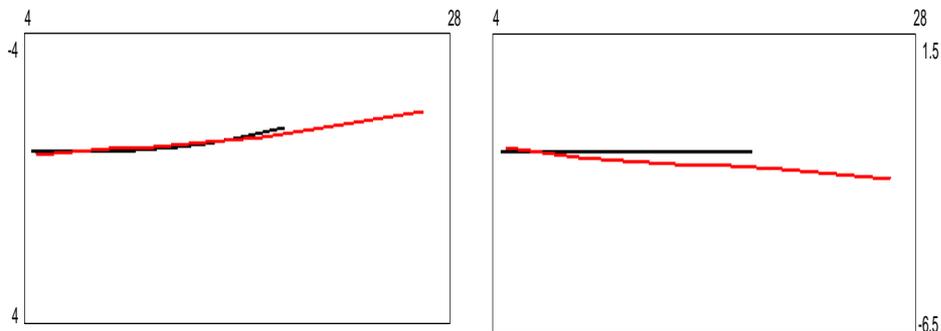
**Approximation der Konturen in jedem einzelnen Bild (PNP).** Um den Rechenaufwand für nachfolgende Algorithmen weiter zu reduzieren und damit den Echtzeitanforderungen der Anwendung zu entsprechen, ist es vorteilhaft, für die jeweiligen Teilkonturen eine möglichst kompakte, datenreduzierende Repräsentation bereitzustellen (Quadratmittel-Passung der ausgewählten Kontursegmente mit Polygonzügen und konischen Splines vorgegebenen Fehlermaßes).

**Bildübergreifende Zuordnung von Polygonsegmenten (PCM).** Für die 3D-Rekonstruktion von Konturen ist die Kenntnis ihrer zeitlichen Korrespondenz erforderlich. Dazu werden die extrahierten und vorgepaßten Kontursegmente in aufeinander folgenden Bildern einander zugeordnet und daraus sogenannte Konturstapel generiert.

Da Kontursegmente durch die Bewegung des Fahrzeugs, die zeitliche Veränderung der Szene, Verdeckungen und Digitalisierungseffekte des öfteren aufbrechen bzw. wieder verschmelzen, muß dafür Sorge getragen werden, daß auch solche Effekte zufriedenstellend gehandhabt werden. Dies impliziert, daß einem momentan betrachteten Polygonsegment im aktuellen Halbbild ein oder mehrere Segmente im vorhergehenden bzw. folgenden Halbbild zugeordnet werden können.

**3D-Rekonstruktion von Konturen (P3D/R3D).** Die quantitative Nutzung der Orientierungskinetik des Bildsensor-Trägerfahrzeugs, sowie der Kalibrierungsdaten der Kamera, erlaubt das Wiederauffinden beliebiger Punkte einer Kontur über mehrere Frames hinweg.

Unter Nutzung der Tatsache, daß sich ortsfeste Objekte im Bild auf Strahlen vom Fluchtpunkt wegbewegen, kann bei bekannter Fahrzeugkinematik für jeden Punkt einer Kontur die Lage seines Sichtstrahls für das nächste Bild berechnet und darüberhinaus durch den Schnitt des projizierten Strahls mit dem korrespondierenden Konturzug ebendieser Punkt wiederaufgefunden werden. Durch wiederholte Anwendung dieses Prinzips "künstlicher" Korrespondenzen, können beliebige Punkte eines Polygonzugs über mehrere Bilder hinweg verfolgt und ihnen dabei räumliche Tiefeninformation zugeordnet werden.



**Abbildung 12.** Originalbild (oben) und Vergleich der 2D-Reprojektion (kurze Linien) mit der berechneten 3D-Rekonstruktion aus fünf perspektivischen Bildern (lange Linien) am Beispiel der Mittelstreifenkontur. Linkes Bild: Draufsicht; rechtes Bild: Seitenansicht; Angaben in Meter

Abb. 12 zeigt als Beispiel Ergebnisse der Rekonstruktion eines Teils der Mittelstreifenkontur einer Landstraßenszene. Da es sich hierbei um eine im wesentlichen quasistationäre Kurve handelt - man erhält trotz Translationsbewegung der Kamera nahezu identische perspektivische Bilder der Kontur - führen Ansätze ohne Verwendung von Zusatzwissen über die Eigenbewegung der Kamera oder

auch solche, die lediglich auf dem optischen Fluß basieren, notwendig zu unbefriedigenden Ergebnissen.

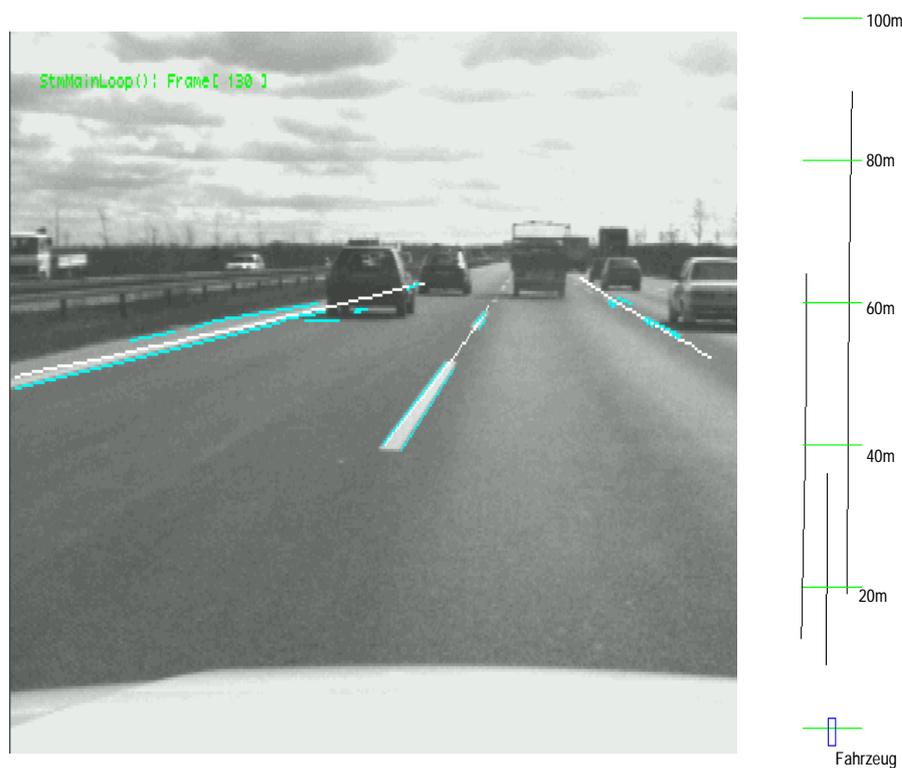
In Abb. 12 dargestellt sind die jeweiligen 3D-Konturen, die man einerseits durch Reprojektion in die Straßenebene (kurze Linie), andererseits mit Hilfe des oben beschriebenen Verfahrens unter Nutzung künstlicher Korrespondenzen bei bekannter Eigendynamik (lange Linie) erhält. Für das gezeigte Beispiel wurden punktweise zugeordnete Konturen aus fünf aufeinanderfolgenden Bildern verwendet. Dies entspricht der zeitlichen Entwicklung der Kontur innerhalb einer  $\frac{1}{10}$  Sekunde.

In der Seitenansicht ist deutlich die Fehlschätzung der räumlichen Ausdehnung der Kontur bei der 2D-Reprojektionsmethode erkennbar. Dies liegt an der perspektivischen Verkürzung, die durch die Fahrt über den leichten Hügel auftritt. Die 3D-rekonstruierte Kontur hingegen ist deutlich länger und verläuft bezüglich der augenblicklichen Standebene des Fahrzeugs nach unten, was die Konvexität des Hügels widerspiegelt.

Auch in der Draufsicht ist die Verkürzung der reprojizierten Kontur deutlich erkennbar. Deren Nichtberücksichtigung führte bei der modellgestützten Einordnung von Hindernissen auf der Straße einerseits zu einer Fehlschätzung des Abstands, andererseits aber auch zu einer falschen Interpretation der Krümmung der Straße, die für die Querführung des Fahrzeugs von entscheidender Bedeutung ist.

**Modellierung des Straßen- und Spurverlaufs (STM).** Die aus den oben beschriebenen Bildverarbeitungsmodulen vorverlesenen Konturen werden mittels Heuristiken den Fahrspuren zugeordnet, welche die Position und Lage, Richtung und die Zusammenhangseigenschaften der Konturen einerseits und geometrische Eigenschaften des aktuellen Spurmodells andererseits als Entscheidungsparameter verwendet. Die einer Spur zugeordneten Konturen werden durch lineare Approximationstechniken an periodische B-Splines gepaßt. Jedes Konturstück trägt dabei nur einen sehr begrenzten Zeitraum (1 - 5 Bilder) zur Spline-Repräsentation bei. Die verwendeten B-Splines bestehen aus einer variablen Anzahl von Segmenten und sind vom Grad 3 oder 4. Dies sichert dem Straßenmodell die Flexibilität, die zur Passung aller in der Realität vorkommenden Straßenkonturen erforderlich ist. Ein beispielhaftes Ergebnis der Straßen- und Spurerkennung ist in Abb. 13 gezeigt. Aus dem errechneten Modell werden die *Regions-Of-Interest* der erkannten Straßenberandungen, sowie Aufmerksamkeitsbereiche weiterer potentieller Spurberandungen bestimmt. Diese werden als Anforderung an das Modul zur Polygonzugselektion (PNP) rückgekoppelt.

**Bildkantenbasierte Detektion und Verfolgung verkehrsrelevanter Hindernisse (OTR).** Durch Nachbarschaftsgruppierung horizontaler Kanten der Konturrepräsentation und geeigneten Gruppenstabilisierungen werden verkehrsrelevante Hindernisse detektiert und über die Zeit verfolgt. Dabei findet 3D-Kanteninformation und Fahrzeugbewegungsinformation Verwendung. Das Verfahren zeichnet sich durch kurze Laufzeiten auch bei mehreren verfolgten Ob-



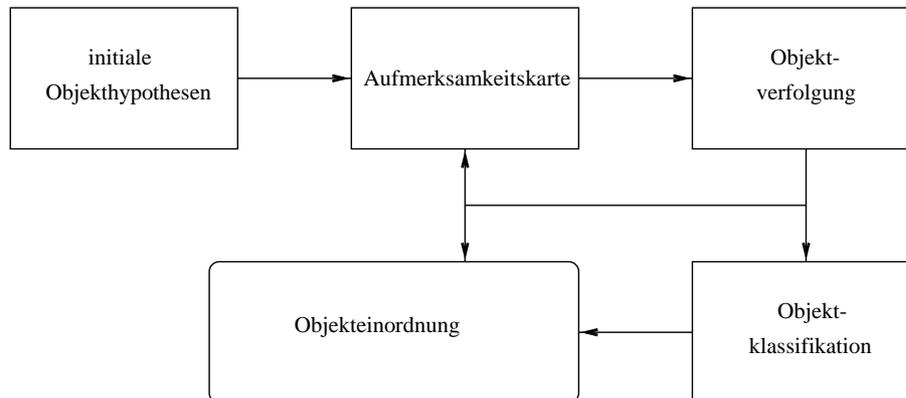
**Abbildung 13.** Kartierung des Straßen- und Spurverlaufs. Links: Originalszene mit Spurberandungen; rechts: Spurbegrenzungen in Draufsicht. Man beachte den interpolierenden Effekt der Passung (linke Fahrbahnbegrenzung).

jekten aus. Auf diese Weise kann sehr schnell die Information über die in der Fahrzeugumgebung vorhandenen Hindernisse ermittelt werden (*primal scetch*).

## 9 Bildpixelbasierte Detektion, Verfolgung und Klassifikation verkehrsrelevanter Hindernisse (INI)

Zielsetzung des in diesem Abschnitt beschriebenen Teilprojekts ist die Detektion und Klassifikation verkehrsrelevanter Hindernisse.

Es wird ein Ansatz vorgestellt, der zunächst Objekthypothesen aus Bildsequenzen generiert, die Hypothesen nutzt, um bildlokal Objektklassifikationen vorzunehmen, die Objekte über die Zeit verfolgt und das vorausgehend erwähnte Klassifikationsergebnis - über mehrere Frames verteilt - verifiziert bzw. falsifiziert. In Abb. 14 sind die Funktionsblöcke des Moduls zur Hinderniserkennung

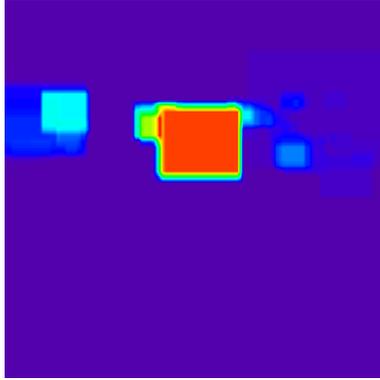


**Abbildung 14.** Hinderniserkennungssystem

schematisch dargestellt. Detektierte Objekthypothesen werden zur Initialisierung, Verifikation und Stabilisierung in einer Aufmerksamkeitskarte abgelegt. Hierbei wird durch Erzeugung von Redundanz die Robustheit der Ergebnisse der Objektverfolgung durch Rückkopplung signifikant erhöht. Die aus der Aufmerksamkeitskarte resultierenden Objekte werden schließlich über die Zeit verfolgt und klassifiziert. Die simultan gewonnenen Objektattribute, wie Größe, Abstand, Geschwindigkeit und Objektklasse stützen den Prozeß bildübergreifend (siehe Abb. 14). In den folgenden Abschnitten werden die einzelnen Prozeßschritte grob skizziert.

**Generierung von Objekthypothesen.** Ein erster Schritt in der Verarbeitungskette ist die initiale Detektion potentieller Hindernisse auf der Fahrbahn. Hierfür stehen etablierte Verfahren zur Verfügung; [17, 18]. In der hier beschriebenen Anwendung beschränkt man sich aufgrund der zur Verfügung stehenden, kurzen Rechenzeiten auf die Detektion von Bildbereichen mit ausgeprägtem, negativ-vertikalen Pixelgradienten und beschränkt diese auf den Ausschnitt der Fahrbahn. Da dieser ikonisch orientierte Verarbeitungsschritt am Beginn der Verarbeitungskette steht und mit wenig Daten- bzw. Modellwissen operiert, darf die Reduktion der gesamten Bildinformation auf Aufmerksamkeitsbereiche nur eingeschränkt vollzogen werden. Durch die nachgeschalteten Module ist eine engere Fokussierung auf die interessierenden Objektbereiche zu leisten.

**Aufmerksamkeitskarte.** Die Bewertung einzelner Bildbereiche in Bezug auf die Aufgabenstellung wird durch eine Aufmerksamkeitskarte realisiert, in welche spezielle Pixelattribute eingetragen werden. In der gewählten Repräsentation werden einkoppelnde Regionen über lokale "Aktivitäten" durch anwendungsspezifisch geeignete Funktionale bewertet und deren Signifikanz verifiziert (vgl. Abb. 15).

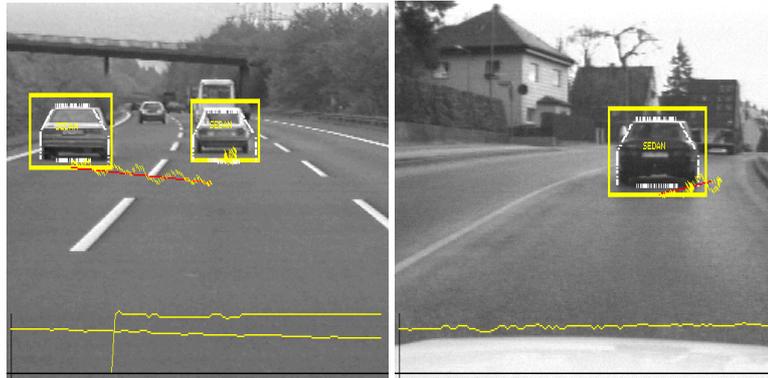


**Abbildung 15.** Aufmerksamkeitskarte am Beispiel der Objektverfolgung

Die Integration verschiedener Bewertungsmodule in das Gesamtsystem wird durchgeführt [19]. Module zur Sensordatenfusion [16] können auf dieser Verarbeitungsstufe in Zukunft ebenfalls integriert werden. Zur zeitlichen Stabilisierung und Verbesserung der Robustheit, führt die Aufmerksamkeitskarte die Zeit als dritte Achse mit und stellt nachfolgenden Verarbeitungsmodulen eine Liste interessierender Bildbereiche zur Verfügung, deren Hindernissignifikanz abfragbar ist. Die Einkopplung in die Objektverfolgung und -klassifikation ist realisiert. Eine Rückkopplung in die Objektdetektion ist prinzipiell möglich und in Zukunft geplant.

**Objektverfolgung.** Bei der Objektverfolgung ist der geometrische Vergleich von Gestaltsignaturen die zu lösende Hauptaufgabe. In dem gewählten Ansatz [21] wird die Abweichung eines Satzes von Modellmerkmalen zu den Bildmerkmalen mit Hilfe der *Hausdorff-Distanz* gemessen. Hierbei wird zum Zeitpunkt  $t = 0$  ein transientes Modell des Objektes akquiriert und über der Zeit unter den Freiheitsgraden der Translation und Skalierung in der Bildsequenz wiedergefunden. Um ein möglichst hohes Maß an Robustheit und Flexibilität zu gewährleisten, wird das Modell anhand der wiedergefundenen Bildmerkmale über die Zeit adaptiert. Hierbei ist ein Verfahren entwickelt worden, welches gestattet das Modell objektselektiv anzupassen, um eine Adaption an Hintergrundbereiche auszuschließen. Der Algorithmus ist auf einer Vielzahl von Bildfolgen unterschiedlicher Geometrie und unterschiedlicher Beleuchtungsverhältnisse erfolgreich getestet worden (Abb. 16).

**Objektklassifikation.** Bei der Hindernisklassifikation kommen in erster Linie Neuronetz-basierte Methoden zum Einsatz [20, 21]. Grundsätzlich ist zwischen merkmals- und modellbasierten Klassifikatoren zu unterscheiden. Der schnelle merkmalsbasierte neuronale Klassifikator führt mit Hilfe der lokalen Orientierungskodierung [21] eine Initialklassifikation von Fahrzeugen durch. Zeilen-



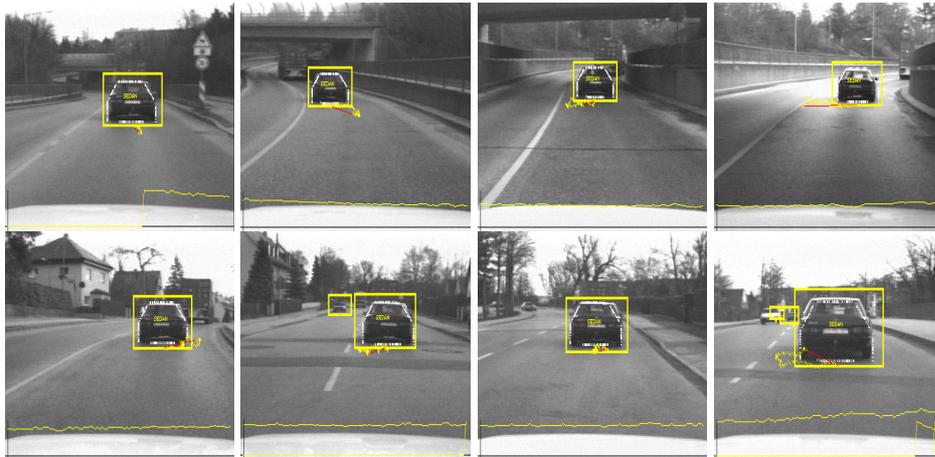
**Abbildung 16.** Objekt-detektion, -klassifikation und Objektverfolgung auf Autobahnen und Landstraßen

und Spaltenhistogramme werden als Eingangsdaten einem neuronalen MLP-Netz (universeller Approximator kontinuierlicher Funktionen) übergeben. Der Netzausgang beschreibt die Klasse der übergebenen Regionen. Im konkreten Fall wurden drei Klassen (Pkw, Lkw, und Transporter), sowie der Hintergrund gelernt. Der modellbasierte neuronale Klassifikator ordnet Bildregionen mit Hilfe der Hausdorff-Distanz [21] statisch festgelegten Modellen (hier: Limousine, Lieferwagen und Lkw) zu. Ein neuronales Netz bewertet hierbei die Ähnlichkeit zwischen den Modellen und der gegebenen Hypothese. Beide Typen Klassifikatoren wurden unter Verwendung einer Datenbank mit 1500 verschiedenen Objekten, die mit unterschiedlichen Kameratypen und -geometrien aufgenommen wurden, trainiert.

## 10 Ergebnisse und Diskussion; Ausblick

Zielsetzung des Projektes war die Entwicklung eines prototypischen, hochsprachenprogrammierbaren Systems zur flächendeckenden Interpretation von Bildfolgen für eine schwierige Klasse von Anwendungen. Dabei waren erhebliche Zielkonflikte hinsichtlich der Leistungsfähigkeit, Herstellungskosten, Baugröße, Umweltverträglichkeit und Verlustleistung der zugrundeliegenden Hardware zu bewältigen. Um den entstandenen Versuchsträger einem größeren Kreis (in der Regel wenig hardware-orientiert denkender Software-Entwickler) zuzuführen, wurde das System mit einer leistungsfähigen SW-Entwicklungs- und Produktionsumgebung versehen. Die damit geschaffenen Möglichkeiten ermöglichen nicht nur kürzeste Entwicklungszyklen (Rapid Prototyping) komplexer, vollbildauswertender Systemanwendungen in Software, sondern erlauben erstmals deren unmittelbar folgende schritthaltende, experimentelle Erprobung im Feld.

Dieser Aspekt ist insofern hoch zu bewerten, als die überwiegende Anzahl der in der Literatur veröffentlichten Vollbildverfahren typischerweise auf nur weni-



**Abbildung 17.** Objektdetektion, -klassifikation und Objektverfolgung an einer Sequenz von 1000 Bildern mit unterschiedlichen Beleuchtungsverhältnissen

gen Bildern oder Bildfolgen erprobt werden und die zugrundeliegenden Verfahren daher erfahrungsgemäß häufig auf die darin enthaltenen Testmuster "trainiert" werden. Da es darüberhinaus keine geschlossene "Systemtheorie des Bildverstehens" gibt, ist man hinsichtlich der Funktions- und Robustheitsuntersuchung prototypischer Gesamtanwendungen notwendig auf umfangreiche experimentelle Untersuchungen angewiesen. Dies gilt zumindest für die hier vorgestellten Applikationen zur Fahrerassistenz und die ihnen eigene immense Szenenvariabilität.

Experimentelle Untersuchungen zu den in diesem Artikel vorgestellten Verfahren erfordern bereits im Open-Loop-Betrieb Echtzeitfähigkeit. In 1994 hätte die beaufsichtigte Verrechnung einer Stunde Bildmaterial auf dann verfügbarer Workstation-Hardware ca.  $\frac{1}{2}$  Mannjahr benötigt.

Mit der im Projekt entstandenen Entwicklungs- und Versuchsplattform wurde erstmals die Basis dafür geschaffen, *Computer Vision*-Anwendungen der hier beschriebenen Komplexität, ernsthaft dem Experiment zuzuführen. Die Leistungs- und Systemanforderungen der oben charakterisierten Entwicklungs- und Versuchsplattform wurden u.a. von denen der vorgestellten prototypischen Bildverarbeitungsanwendungen abgeleitet. Die Echtzeitfähigkeit der realisierten Verarbeitungsprozesse zur Eigenbewegungsbeobachtung, zur schritthaltenen Kartierung des Straßen- und Spurenverlaufs und zur Hindernisdetektion ist daher im wesentlichen bereits gesichert. Zeitgleich zur Verfassung dieses Artikels beginnt die Systemintegration der genannten, im Off-line-Betrieb extensiv getesteten Prozesse. Erste Feldversuche sind noch für das 2. Halbjahr 1998 geplant.

Kurz- und mittelfristige Anschlußprojekte beinhalten neben algorithmischen Verfahrensverbesserungen:

- die Kopplung der in Abschnitt 2 beschriebenen CMOS-Kamera mit der SAP-Baugruppe
- die Entwicklung höher auflösender CMOS-Sensorkomponenten (Ziel: bis zu  $1024 \times 512$  Pixel)
- die weitere Miniaturisierung der SAP-Hardware (unter Erhöhung der Taktfrequenz)
- die Entwicklung von Spin-Off Anwendungen deutlich reduzierter Leistungsanforderungen, ggf. auch: Integration intermediärer Bildverarbeitungsprozesse

Die in diesem Beitrag beschriebenen Module, Systeme und Anwendungen werden auf der vom 29.09. bis 01.10.1998 zeitgleich zum DAGM Symposium stattfindenden "Vision 98" präsentiert.

## 11 Danksagung

Die Autoren bedanken sich zunächst beim Bundesministerium für Forschung und Technologie (BMBF) und dem eingeschalteten Projektträger für Informationstechnologie für die Förderung des Vorhabens unter den Kennzeichen IT-ME-01M3010A5, IT-ME-01M3010B und IT-ME-01M3010C0. Das mit der Verbundleitfunktion betraute BMW Projektteam dankt den zuständigen Entscheidungsträgern der Firmen BMW AG und BGT GmbH für die Bewilligung des Vorhabens und der dafür erforderlichen Ressourcen, die das Projekt und die damit erzielten Ergebnisse im Grundsatz ermöglicht haben. Besonderer Dank gilt den Projektleitern und Entwicklungsteams der Verbundpartner für ihr über die Projektlaufzeit aufrechterhaltenes großes Engagement und ihren unermüdlichen Arbeitseinsatz.

Aufgrund der standort- und institutionsübergreifend engen Zusammenarbeit von zwanzig bis dreißig zumindest zeitweise im Projekt arbeitenden doktoranden und Entwicklungsingenieuren, haben wir auf die Angabe einer Autorenliste verzichtet. Die Namen der hauptsächlich zum Projekt beitragenden Personen können über die Veröffentlichungen [1–3, 22, 14, 15, 21] zugeordnet werden.

## Literatur

1. M. Baur, T. Schumm, R. Wertheimer: Abschlußbericht BMW zu BMBF Förderkennzeichen IT-ME-01M3010A5. Technische Informationsbibliothek (TIB) (1998), Postfach 6080, D-30060 Hannover, FRG
2. Abschlußbericht BGT zu BMBF Förderkennzeichen IT-ME-01M3010B. Technische Informationsbibliothek (TIB) (1998), Postfach 6080, D-30060 Hannover, FRG
3. M. Schanz, T. Eckart, Ch. Nitta, B.J. Hosticka, R. Wertheimer: CMOS Imager with 120 dB Linear Output. Submitted for publication in Proceeding of the ISSCC 99
4. C.G. Harris, M.J. Stephens: A combined corner and edge detector. in: Proceedings, 4th Alvey Vision Conference (1988) 147–152
5. A. Mitiche, P. Bouthemy: Computation and Analysis of Image Motion: A Synopsis of Current Problems and Methods. International Journal Computer Vision, **19** (1996) 29–55

6. Q.T. Luong, O.D. Faugeras: The Fundamental Matrix: Theory, Algorithms and Stability Analysis. *International Journal Computer Vision*, **17** (1996) 43–75
7. J. Weng, T. S. Huang, N. Ahuja: *Motion and Structure from Image Sequences*. Springer-Verlag, Berlin, Heidelberg (1993)
8. Z. Zhang, O. Faugeras: *3D Dynamic Scene Analysis*. Springer-Verlag, Berlin, Heidelberg (1992)
9. L. S. Shapiro, A. Zisserman, M. Brady: 3D Motion Recovery via Affine Epipolar Geometry. *International Journal of Computer Vision*, **16** (1995) 147–182
10. Q.-T. Luong, O.D. Faugeras: Self-Calibration of a Moving Camera from Point Correspondences and Fundamental Matrices. *International Journal Computer Vision*, **22** (1997) 261–289
11. J.M. Orwell, J.F. Boyce, J.F. Haddon: Ego Motion from Near-Degenerate Sequences. in: *Proceedings, 13th International Conference on Pattern Recognition, Vienna, Austria (1996)* 412–416
12. T. Viéville, O.D. Faugeras, Q.T. Luong: Motion of Points and Lines in the Uncalibrated Case. *International Journal Computer Vision*, **17** (1996) 7–41
13. R. Wagner, K. Donner, F. Liu: A "Half-Perspective" Approach to Robust Ego-Motion Estimation for Calibrated Cameras. *Fakultät für Mathematik und Informatik, Universität Passau, MIP-9718 (1997)*
14. R. Wagner, F. Liu, K. Donner: Robust Motion Estimation for Calibrated Cameras from Monocular Image Sequences. *erscheint in: Computer Vision and Image Understanding*
15. J. Haas: *Echtzeit-Korrespondenzprobleme in Bildsequenzen und Subpixelgenauigkeit*. Dissertation, Universität Passau, Fakultät für Mathematik und Informatik (1998)
16. U. Handmann and G. Lorenz and W. v. Seelen: Fusion von Basisalgorithmen zur Segmentierung von Strassenverkehrsszenen. in *Mustererkennung 1998 (1998)*, Springer-Verlag, Berlin, Heidelberg
17. T. Kalinke and C. Tzomakas: Objekthypothesen in Verkehrsszenen unter Nutzung der Kamerageometrie. *IRINI 97-07 apr (1997)*, Institut für Neuroinformatik, Ruhr-Universität Bochum
18. U. Handmann and T. Kalinke: Fusion of texture and contour based methods for object recognition. *ITSC'97 (1997)*
19. U. Handmann and G. Lorenz and T. Schnitger and W. von Seelen: Fusion of different Sensors and Algorithms for Segmentation. *ITSC'98*
20. T. Kalinke and C. Tzomakas and M. Werner and W. v. Seelen: Texture-Based Object Detection and an Adaptive Model-Based Classification. *ITSC'98*
21. U. Handmann and T. Kalinke and C. Tzomakas and M. Werner and W. v. Seelen: *Computer Vision for Driver Assistance Systems*. *Proceedings of SPIE Vol. 3364*
22. O. Ehrmann, K. Buschick, G. Chmiel, A. Paredes, V. Glaw, H. Reichl: 3D - Multichip Module, *Proceedings International Conference on Multichip Modules, Denver, 1995*, pp.358-363
23. J. Wolf, F.J. Schmöckle, W. Heinrich, M. Töpfer, K. Buschick, A. Owzar, O. Ehrmann, H. Reichl: System Integration for High Frequency Applications, *Proceedings International Symposium on Microelectronics, Philadelphia, 1997*, pp.29-36
24. M. Töpfer, K. Buschick, J. Wolf, V. Glaw, R. Hahn, O. Ehrmann, H. Reichl: *Embedding Technology - A Chip-First Approach using BCB, Advancing Microelectronics, Vol.24, No.4 (1997)* pp.18-21