



Jan Deiterding, Dr. Dominik Henrich

## Die Sensorik-Integration

Mit dem Projekt „A-Bot“ haben Forscher der Universität Bayreuth einen neuen Ansatz entwickelt, über den sich externe Sensorik wie etwa Kameras auf einfache Art in Roboterprogramme integrieren lassen. Ein Konzept, welches insbesondere kleinen und mittelständischen Unternehmen einen wirtschaftlichen Einsatz von Robotern für Handhabungsaufgaben ermöglichen soll.

**B**ei der Programmierung eines Roboters für industrielle Anwendungen sind zwei Kriterien entscheidend: Zum einen die Zeitdauer für die Entwicklung eines Roboterprogramms, welches die gestellte Aufgabe verlässlich löst (Entwicklungszeit). Zum anderen, wie lange der Roboter im Durchschnitt dazu braucht, um die gestellte Aufgabe auszuführen (Ausführungszeit). In der Großindustrie ist letzteres von primärem Interesse. Um angesichts der dort vorherrschenden Stückzahlen, die bis in die Millionen gehen können, die Ausführungszeit zu optimieren, nehmen die Hersteller eine lange Entwicklungszeit in Kauf. Dies rührt unter anderem daher, dass zur Minimierung der Standardabweichung im Allgemeinen spezielle Zuführeinrichtungen angefertigt

werden. Die Idee dahinter ist, Toleranzen und Lage-ungenauigkeiten von Objekten zu kompensieren, so dass der Roboter die Aufgabe nahezu ohne den Einsatz externer Sensoren, wie beispielsweise Kameras oder Kraft-/Momentsensoren, lösen kann. Dies deshalb, weil eine Sensordatenverarbeitung relativ zeitaufwendig ist.

Anders stellt sich die Situation in kleinen und mittleren Unternehmen dar: Ein Roboter soll hier in der Lage sein, eine Vielzahl von Aufgaben mit möglichst geringem Umrüst-Aufwand durchzuführen. Der Anspruch besteht hier eher darin, den Menschen als Arbeiter zu entlasten. Nur dann, wenn der Roboter – inklusive der Entwicklungszeit – die Aufgabe zumindest gleich schnell wie ein Mensch ausführt, ist ein Einsatz rentabel.

Bei Bearbeitungsaufgaben, wie zum Beispiel Schweiß- oder Lackieraufgaben, muss der Roboter keine Entscheidungen treffen, sondern nur eine vorgegebene Bahn mit einer festen Geschwindigkeit abfahren. Zur Programmierung solcher Bahnen existiert inzwischen eine Reihe komfortabler Software-Umgebungen.

Was Handhabungsaufgaben betrifft, ist die Ausgangslage eine andere: Sie unterscheiden sich von Bahnplanungsaufgaben in dem Sinne, dass hier Objekte, die in Lage und Orientierung variieren können, miteinander in Kontakt gebracht werden müssen. Darüber hinaus sind viele Handhabungsaufgaben nicht trivial. Das Einfügen eines Objektes in ein anderes Objekt mit nur geringen Toleranzen erfordert eine große Anzahl von Fallunterscheidungen auf Seiten des Roboters. Da der Entwurf von Zuführeinrichtungen, wie sie in der Großserienfertigung üblich sind, für diese Anwenderklientel aus Kostengründen meist nicht in Frage kommt, ist die Verwendung externer Sensoren hier die einzige Möglichkeit, um ein breites Anwendungsspektrum eines Roboters zu gewährleisten. Dies erhöht jedoch die Komplexität der Programmierung, da neben den

(Bild: Kuka)

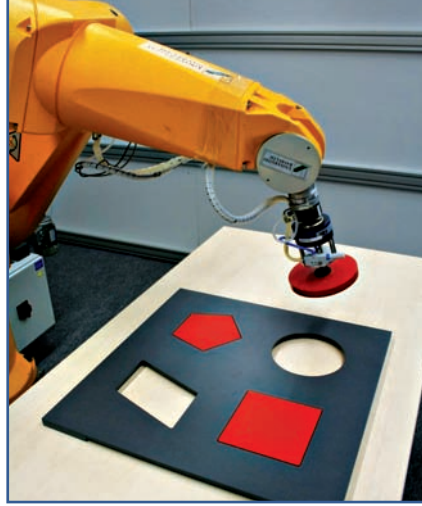
Instruktionen zur Bewegung des Roboters nun auch Kenntnisse in Sensordatenverarbeitung erforderlich sind. Gerade KMUs beschäftigen aber nur selten Experten mit ausreichendem Fachwissen in beiden Bereichen. Kurzum: Um Roboter für diese Unternehmen interessant zu machen, gilt es daher, die Programmierung von Robotern mit externer Sensorik stark zu vereinfachen – und zwar unabhängig sowohl von der Art des verwendeten Sensors als auch von der eigentlichen Aufgabe.

An dieser Stelle setzt das Projekt A-Bot des Lehrstuhls für Robotik und eingebettete Systeme der Universität Bayreuth an, in dessen Rahmen ein zweistufiges Entwicklungskonzept modelliert wurde:

Auf der ersten Stufe wird das Grundgerüst der Aufgabe festgelegt. Dieses beschreibt die einzelnen Arbeitsschritte des Roboters in sequenzieller Reihenfolge inklusive Verzweigungen und Wiederholungen. Die Aufgabe für den Entwickler besteht in diesem Teil im Wesentlichen darin, Roboterbewegungen zu generieren und miteinander zu verbinden. Dies kann sowohl grafisch als auch textuell geschehen. Sofern alle Objekte der Aufgabe keinen Toleranzen unterworfen sind, existiert damit bereits ein lauffähiges Roboterprogramm, welches in der Lage ist, die Aufgabe korrekt auszuführen. Das Programm unterteilt sich hierbei in eine Reihe von Instruktionen, welche die Durchführung der Aufgabe beschreiben, sowie in eine Liste von Positionen, die den Ort der Durchführung festlegen.

**Sensordatenverarbeitung vom Roboterprogramm entkoppelt**

Auf der zweiten Stufe wird die benötigte Flexibilität gewonnen, indem die Integration externer Sensoren in das Programm erfolgt. Auf diese Weise lassen sich Toleranzen in Bezug auf die Abmessungen und die Lage der Objekte kompensieren. Allerdings sind hierfür umfangreiche Veränderungen des Programms erforderlich.

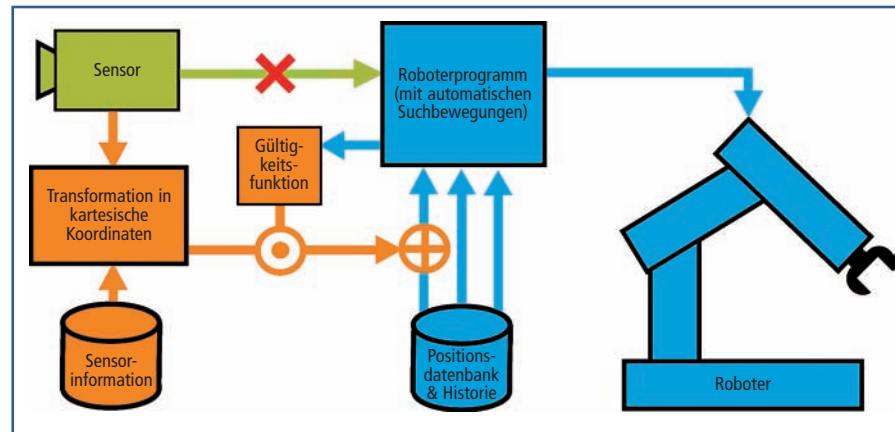


**Bei Einfüge-Operationen mit nur geringen Toleranzen müssen Roboter externe Sensoren verwenden, um die Aufgabe erfolgreich lösen zu können. Die Berechnung der Roboterbewegungen aufgrund der gemessenen Sensordaten ist diffizil und erfordert Spezialwissen und eine hohe Entwicklungszeit.**

Normalerweise fügt ein Entwickler jetzt die notwendigen Instruktionen zur Sensordatenverarbeitung direkt in das Programm ein. Dies bringt jedoch diverse Nachteile mit sich: Erstens wird das Programm unübersichtlich, weil es zu einer Vermischung der Bewegungsbefehle für den Roboter mit Rechenbefehlen zur Sensorauswertung kommt. Eine Wartung oder Modifikation zu einem späteren Zeitpunkt

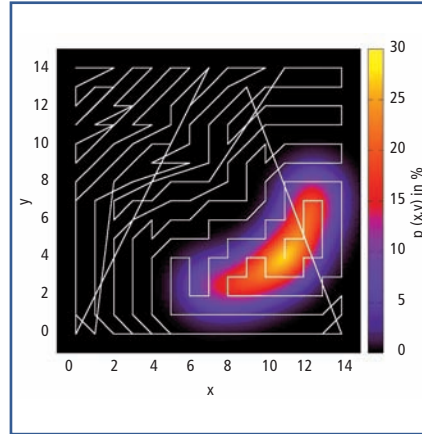
würde sich aus diesem Grund sehr aufwendig gestalten. Zweitens muss der Entwickler genau wissen, an welcher Stelle im Programmfluss die Sensordaten aufzunehmen und auszuwerten sind. Und drittens muss die Sensorinformation oft durch den Entwickler aufbereitet werden. Dazu gehört beispielsweise eine Rauschunterdrückung oder eine Bildverarbeitung bei Kameras. Dies setzt ein umfangreiches Wissen beim Entwickler voraus und erschwert die Austauschbarkeit der Sensoren. Mit anderen Worten: Ein Programm ist nun stark an den verwendeten Sensor gekoppelt und nicht ohne weiteres auf ein anderes System übertragbar.

Da im Bereich der industriellen Handhabung externe Sensoren fast ausschließlich zur Identifikation von Abweichungen zum Einsatz kommen, werden die Sensorinformation nur dazu verwendet, eine bereits bekannte Position zu modifizieren – nicht aber, um eine neue Position zu berechnen. Aus diesem Grund wurde im Rahmen von A-Bot ein Konzept der Sensorintegration entwickelt, in dem die Sensordatenverarbeitung getrennt vom Roboterprogramm erfolgt.



**In einem positionsorientierten Programmierkonzept erfolgt die Auswertung der Sensorinformation nicht direkt im Roboterprogramm, sondern wird erst in kartesische Koordinaten transformiert. Nach einer Gültigkeitsprüfung werden die Koordinaten auf Positionen in der Positionsdatenbank angewandt. Für den Roboter ist nicht erkennbar, dass eine Position von einem Sensor abhängig ist.**

<p><b>Edelstahl</b></p>	<p><b>Edelstahl</b></p>	<p><b>TWK</b>  <b>TWK-ELEKTRONIK</b>  <b>Winkelcodierer</b>  <b>Wegaufnehmer</b>          T. +49/211/632067 F. +49/211/637705          www.twk.de info@twk.de</p>	<p><b>Edelstahl</b></p>	<p><b>Edelstahl</b></p>
-------------------------	-------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------	-------------------------



**Beispiele für Suchbewegungen in der Ebene. Auf dem linken Bild fährt der Roboter bei einer einfachen Suche in einer spiralförmigen Bewegung die Ebene ab, bis er das Ziel gefunden hat. Auf dem rechten Bild ist eine Wahrscheinlichkeitsverteilung für eine intelligente Suche gezeigt. Der Roboter merkt sich erfolgreiche Positionen aus vergangenen Ausführungen und passt den Suchpfad (weiß dargestellt) so an, dass die durchschnittliche Suchzeit in der aktuellen Ausführung minimiert wird.**

Anstelle der Auswertung der Sensorinformation im Programm selbst, findet bei diesem Ansatz die Modifizierung der Positionen direkt durch die Sensoren statt. Dies deshalb, weil die Sensorinformation über ein Objekt nur noch eine Veränderung seiner Position beschreibt. Es ist jedoch keine Planung mehr erforderlich, die eine grundlegende Veränderung des Programmablaufes bedingen würde – eine Fehlerbehandlung einmal ausgenommen. Kommt es während der Ausführung zu einem Zugriff auf die Position, so wird der Sensor ausgewertet und die Ausgangsposition entsprechend modifiziert. Das Entscheidende dabei: Für den Roboter ist zu keinem Zeitpunkt erkennbar, dass diese Position sensorabhängig ist.

Um von einem bestimmten Sensortyp zu abstrahieren, erfolgt eine explizite Transformation des Sensorsignals in eine kartesische Positionsveränderung. Diese Transformation ist sowohl von der Aufgabe als auch vom verwendeten Sensor abhängig und dementsprechend vom Entwickler an-

zugeben. Bei einfachen Sensoren – beispielsweise Distanzsensoren – ist dies in der Regel trivial. Für komplexe Sensoren, wie etwa Kameras, existieren bereits industrielle Lösungen, um diese Transformationen automatisiert bestimmen zu können.

Um eine Position entsprechend zu erweitern, muss der Entwickler nun lediglich einige zusätzliche Information in der Positionsbeschreibung angeben: Den Sensor, der die Position überwacht beziehungsweise modifiziert; eine Angabe, wie das Sensorsignal die Position verändert und das Koordinatensystem, in dem diese Transformation stattfindet. Es findet jedoch keinerlei Veränderung im eigentlichen Programm statt.

Bei dieser Realisierung kann es allerdings noch zu Verdeckungsproblemen kommen. Sprich, der Sensor liefert falsche Informationen aufgrund der aktuellen Roboterposition in Bezug auf den Sensor. Dies kann zum Beispiel auftreten, wenn bei Verwendung einer Kamera der Roboter zwischen die Kamera und das zu überwa-

chende Objekt fährt. In diesem Falle ist eine Lokalisierung nicht mehr möglich. Ein anderes Beispiel ist die Verwendung eines Kraft-/Momentsensors am Werkzeugflansch des Roboters. Hier sind nur dann realistische Werte messbar, wenn der Roboter in Kontakt mit einem Objekt ist und sich nicht bewegt. Andernfalls besitzen die gemessenen Werte keine Aussagekraft.

Um diese Problematik zu überwinden, verwendet man Gültigkeitsfunktionen. Mit einer Gültigkeitsfunktion markiert der Entwickler im Programm, ob ein Sensor zu diesem Zeitpunkt gültige Werte liefert oder nicht. Wird nun eine Position aufgerufen, die von einem dieser Sensoren abhängt, so wird der letzte gültige – und nicht der aktuellste! – Wert zur Berechnung verwendet. Durch die Gültigkeitsfunktionen kommt es also zu einer Kopplung des zeitlichen Programmablaufes mit der Sensorauswertung.

### Suche anhand von Karten

Wird ein Sensor „vorlaufend“ eingesetzt – spricht die Auswertung ist nicht von einer Roboterbewegung abhängig –, so lassen sich mit diesem Konzept bereits intuitiv Roboterprogramme erstellen, die auf ihre Umwelt reagieren können. Bei einer „mitlaufenden“ Sensorik, wie sie vor allem für Suchbewegungen typisch ist, hängt die nächste Roboterbewegung jedoch vom aktuellen Sensorwert ab. Bei einer „blinden“ Suche kann der Roboter über den Sensor nur die binäre Entscheidung treffen, ob das Ziel gefunden wurde oder nicht. Beispielhaft hierfür ist die Lokalisierung eines Loches in einer Platte über einen Kraft-/Momentsensor. Der Roboter fährt in diesem Fall einen starren Suchpfad entlang der Oberfläche ab, bis durch einen Knick im Kraft-/Momentsignal das Loch erkannt wird. Bei einer „informierten“ Suche hingegen kann der Roboter anhand der Sensorinformation auch Rückschlüsse darauf ziehen, welcher Punkt im Suchraum als nächstes angefahren werden soll. Beispielhaft hierfür sind Einfüge-Operationen von Objekten in passende Löcher. Anhand der gemessenen Kräfte kann der Roboter berechnen, wie er die nächste Einfügebewegung vornehmen muss.

Beide Arten der Suche sind über Karten realisierbar, die den Suchpfad spezifizieren. Wird eine Position derart markiert, fährt der Roboter bei einem Positionsaufwurf die Startposition der Suche an und fährt dann den Suchpfad ab, bis die Suche er-





**Mess-, Regel- und Überwachungsgeräte für Haustechnik, Industrie und Umweltschutz.**

**AFRISO-EURO-INDEX GmbH**  
 Lindenstraße 20 · 74363 Güglingen  
 Tel. 07135/102-0 · Fax 07135/102-147  
 www.afriso.de · info@afriso.de

folgreich beendet ist. Dies geschieht automatisch ohne die Notwendigkeit einer Modifikation des Roboterprogramms, da an dieser Stelle keine echte Planung erforderlich ist. Der Entwickler muss nur die Positionsbeschreibung entsprechend erweitern.

Auf diese Art und Weise ist es nicht nur möglich, sensorgesteuerte Roboterprogramme schnell und intuitiv zu entwickeln, sondern es kann auch ein Adaptionsprozess des Roboters integriert werden. Dies geschieht in Analogie zu einem menschlichen Arbeiter: Bei den ersten Ausführungen wird der Roboter die Aufgabe zwar korrekt lösen, jedoch relativ langsam. Je öfter er die Aufgabe durchführt, desto mehr Erfahrungen hat der Roboter über seine Umwelt gesammelt und kann so die Ausführungszeit minimieren. Dies geschieht beispielsweise durch adaptive Suchpfade. Dabei wird aus allen vergangenen, erfolgreichen Suchpositionen eine Wahrscheinlichkeitsverteilung konstruiert. Das heißt: Der Roboter beginnt die Suche an der wahrscheinlichsten Position und fährt dann entlang dieser Verteilung zuerst wahrscheinliche Punkte an, und erst danach unwahrscheinliche Punkte. Somit reduziert sich die Suchzeit auf ein Minimum.

Zusammenfassend lässt sich festhalten: Das von der Universität Bayreuth entwickelte System ermöglicht KMU, Roboter auch für Handhabungsaufgaben einzusetzen, ohne dafür Spezialisten beschäftigen zu müssen. Der Unterschied zu anderen Programmiersystemen, wie beispielsweise Taskframe-basierten Ansätzen, liegt darin, dass bei A-Bot besonderer Wert auf

eine intuitive Beschreibung der erlaubten Objektvariationen gelegt wird, um Handhabungsaufgaben ohne zusätzliche Planungsalgorithmen auszuführen. Zwar erlauben Taskframe-basierte Ansätze effizientere geregelte Bewegungen entlang von Oberflächen und ermöglichen auch Planungsaufgaben, dies geht jedoch zu Lasten der einfachen Programmierbarkeit, da wieder gute Kenntnisse in Sensordatenverarbeitung und Regelungszyklen erforderlich sind.

Das A-Bot Gesamtsystem wird direkt in die Robotersteuerung integriert, um eine effiziente Datenverarbeitung zu gewährleisten. Hat sich der Entwickler mit dem neuen Konzept vertraut gemacht, kann er direkt komplexe Handhabungsaufgaben entwerfen. Die Umsetzung auf der Robotersteuerung bleibt dabei komplett verborgen. Eine Dissertation zu A-Bots ist gerade im Abschluss. Danach wird industrieller Einsatz angestrebt. *gh*



**Jan Deiterding**

ist wissenschaftlicher Mitarbeiter am Lehrstuhl für Robotik und Eingebettete Systeme an der Universität Bayreuth.



**Dr. Dominik Henrich**

hat den Lehrstuhl für Angewandte Informatik III (Robotik und Eingebettete Systeme) an der Universität Bayreuth inne.

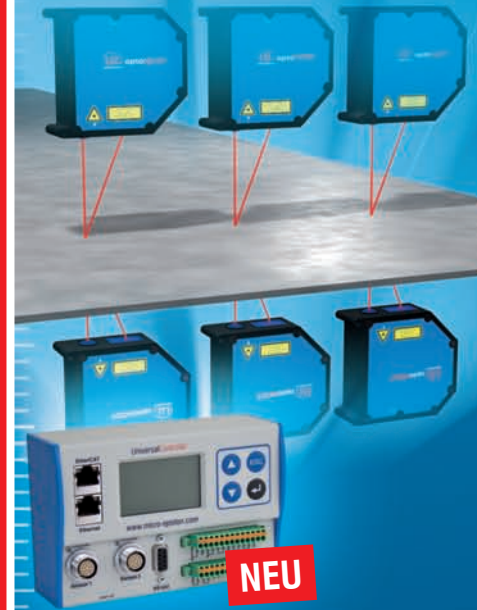
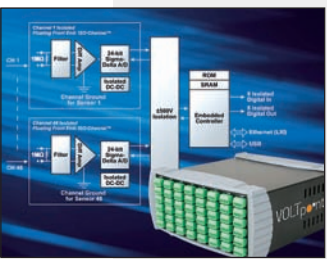
**Messmodule:**

## Keine Angst vorm Rauschen

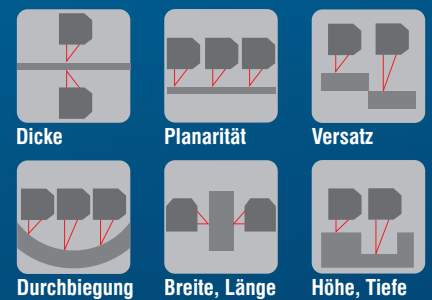
Für genaue Messungen im rauen Industrie-Umfeld hat Data Translation die ISO-Channel-Technologie konzipiert und in die USB- und Ethernet/LXI-Messgeräte der Serien *TEMPpoint* und *VOLTpoint* integriert. Wesentliche Bestandteile des Konzepts sind die galvanische Isolation durch digitale Optokoppler und DC/DC-Wandler mit 24 Bit Auflösung. Dies ermöglicht mikrovoltgenaue Messungen von Spannungen bis 100 V oder von Tem-

peraturen mit Thermoelementen oder Widerstandsthermometern, selbst bei einem Potenzialunterschied von bis zu 1000 V. Erhältlich sind derzeit Geräte mit bis zu 48 galvanisch isolierten Analogeingängen und jeweils acht optoisolierten digitalen E/A. Zum Lieferumfang gehören darüber hinaus eine komfortable Erfassungsoftware für die Visualisierung, Auswertung und Speicherung der Daten sowie verschiedene Treiber, .Net-Klassenbibliotheken und Software-Entwicklungskits.

Data Translation GmbH  
Tel. (0 71 42) 95 31 - 0  
wklass@datx.de  
www.datatranslation.de



## CONTROLLER ZUR DIMENSIONSMESSUNG MIT WEGSENSOREN



- Umfassende Verrechnungsfunktionen
- Für digitale und analoge Sensorsignale
- Automatische Sensorerkennung
- Sehr hohe Verarbeitungsgeschwindigkeit
- Echtzeitfähig

**JETZT INFORMIEREN!**

[www.micro-epsilon.de](http://www.micro-epsilon.de)

MICRO-EPSILON Messtechnik  
94496 Ortenburg  
Tel. 0 85 42/168-0  
info@micro-epsilon.de

**SPS/IPC/DRIVES / Nürnberg**  
24.11. - 26.11.2009  
Halle 7A / Stand 102