

Image-Based Visual Servoing zur autonomen Navigation einer mobilen Plattform

Image-Based Visual Servoing for Autonomous Navigation of a Mobile Platform

Thorsten Lietmann und Boris Lohmann

Ein neuartiger visueller Regler zur autonomen, bildbasierten Navigation einer mobilen Plattform (Rollstuhl) wird vorgestellt. Ziel ist es, den Rollstuhl mittels Image-Based Visual Servoing in eine relative Lage bezüglich einer Landmarke zu navigieren, was allein durch Auswertung von Abweichungen im Kamerabild und ohne Bestimmung der Landmarkenposition in Weltkoordinaten geschieht. Dies wird mithilfe des in diesem Artikel vorgestellten 3D-Visual Servoing erreicht, das eine Erweiterung des herkömmlichen 2D-Visual Servoing darstellt, indem die Entfernung zwischen Kamera und Landmarke mit in die bildbasierte Regelung einbezogen wird. Der visuelle Regler wird dabei durch eine unterlagerte Trajektorienfolge-regelung unterstützt, welche die Fahrt der Plattform überwacht. Zusätzlich wird die Kamera drehbar auf einem Schwenk-Neige-Kopf gelagert, sodass die Landmarke während der Fahrt im Kamerabild gehalten werden kann.

A new visual servoing controller based on the image-jacobian-matrix is proposed and used to navigate a mobile platform (wheelchair). This is done by only evaluating deviations in the camera's image and without using the position of the landmark in the world coordinate frame. The common 2D-visual servoing approach is extended to 3D-visual servoing by adding the distance between landmark and camera into the control scheme. Due to the non-holonomic constraints of the wheelchair the visual servoing has to be supported by an additional trajectory tracking control. Furthermore, a pan-tilt-head is used for turning the camera and keeping the landmark in the camera's field of view.

Schlagwörter: Visual Servoing, mobile Roboter, autonome Navigation, Bild-Jacobi-Matrix, Entfernungsbildkamera, Trajektorienfolgeregelung

Keywords: Visual servoing, mobile robots, autonomous navigation, image-jacobian-matrix, range-image camera, trajectory tracking control

1 Einleitung

Autonome mobile Roboter stellen sowohl im Industriebereich, als auch in der Servicerobotik ein intensiv untersuchtes Forschungsgebiet dar. In diesem Artikel werden Verfahren zur visuellen Regelung einer mobilen Plattform am Beispiel eines Rollstuhls vorgestellt, der Nutzern mit körperlichen Behinderungen eine größere Unabhängigkeit im Alltag ermöglichen soll. Dabei soll der Rollstuhl die Fähigkeiten des Behinderten nutzen und seine nicht mehr vorhandenen Fähigkeiten teilweise ersetzen. Hierbei ist zu beachten, dass die Navigation des Rollstuhls nur mit

sehr allgemeinen Befehlen (z. B. „fahre zum Tisch“ oder „nächste Ecke links“) auskommen sollte. Die Eingabe eines übergeordneten Befehls durch den Benutzer initialisiert also eine teilautonome Fahrt des Rollstuhls.

Um Aktionen von mobilen Robotern in einer dynamisch veränderlichen Umgebung zu ermöglichen, müssen sie in der Lage sein, ihre Umwelt sensorisch zu erkennen. Als interne Sensoren werden hierzu häufig Inkrementalgeber benutzt, welche die Radumdrehungen erfassen (Odometrie) und so unter Zuhilfenahme einer Umgebungskarte eine Positionsbestimmung ermöglichen. Durch den Schlupf der

Räder entsteht jedoch ein Fehler in der Positionsbestimmung, der mit der zurückgelegten Strecke anwächst. Aus diesem Grund und zur Navigation in unbekannter Umgebung werden zusätzlich externe Sensoren eingesetzt. Im hier vorliegenden Anwendungsbeispiel werden eine herkömmliche CCD-Kamera bzw. eine spezielle Entfernungsbildkamera am Rollstuhl montiert. Mithilfe der aus dem Kamerabild gewonnenen Daten ist es möglich, den Rollstuhl in eine relative Lage zu einem Objekt im Raum zu navigieren. Allerdings verhindert die nichtholonome Eigenschaft des Rollstuhls den unmittelbaren Einsatz einfacher Regelverfahren [1]. Dies gilt zunächst auch für das Verfahren des so genannten Image-Based Visual Servoing [2], das ursprünglich für die Regelung von Manipulatoren entwickelt wurde und in diesem Beitrag in verschiedener Hinsicht erweitert wird.

Der am häufigsten gewählte Ansatz zur autonomen Navigation mobiler Roboter verläuft über die Erstellung einer Umgebungskarte. In dieser globalen Karte werden dann Start- und Ziel-Lagen für den Roboter definiert und die Wegplanung auf verschiedene Weisen (z. B. über Graphentheorie [3] oder Trajektorien [4]) in diese Ziel-Lagen realisiert. Dabei wird die Karte entweder offline per Hand [5] oder während einer Erkundungsfahrt vom mobilen Roboter selbst erstellt [6]. Als Sensoren zur Selbstlokalisierung des Roboters während der Fahrt werden Odometriedaten, Kameras, Ultraschallsensoren sowie Laserscanner benutzt und deren gesammelte Daten gegebenenfalls fusioniert [7]. Es existieren auch einige Verfahren, die ohne Umgebungskarte arbeiten: In [8] werden mehrere Kameras und zusätzliche Sensoren zur Merkmalsextraktion benutzt. Das Problem der nichtholonomen Eigenschaft wird in [9; 10] durch die Verwendung einer beweglich an der Plattform angebrachten Kamera umgangen.

In diesem Beitrag werden neue Verfahren der visuellen Regelung mittels einer einzigen Kamera und ohne Umgebungskarte vorgestellt, mit deren Hilfe Ziel-Lagen für den Rollstuhl ermittelt werden können. Dabei handelt es sich um Image-Based Visual Servoing mit herkömmlicher CCD-Kamera und mit spezieller Entfernungsbildkamera, wobei die jeweils verwendete Kamera entweder fest oder drehbar am Rollstuhl montiert ist. Die hierarchische Struktur dieses neuartigen Navigationssystems ist in Bild 1 zu erkennen.

In Kapitel 2 wird zunächst das nichtholonome System „Rollstuhl“ vorgestellt. Die Verwendung einer fest montierten Kamera wird möglich, indem zusätzlich zur visuellen Regelung eine unterlagerte Trajektoriengenerierung benutzt wird, um die Bewegung der Plattform zu planen. Eine zugehörige Trajektorienfolgeregelung sorgt dafür, dass der Rollstuhl auf der geplanten Trajektorie fährt. Kapitel 3 beschreibt das Verfahren des Image-Based Visual Servoing und Modifikationen, die bei Verwendung der Entfernungsbildkamera erforderlich sind. Zusätzlich wird eine Erweiterung des Image-Based Visual Servoing gegenüber bisherigen Ansätzen vorgestellt, indem die Entfernung zwischen Kamera und Landmarke mit in die Regelung ein-

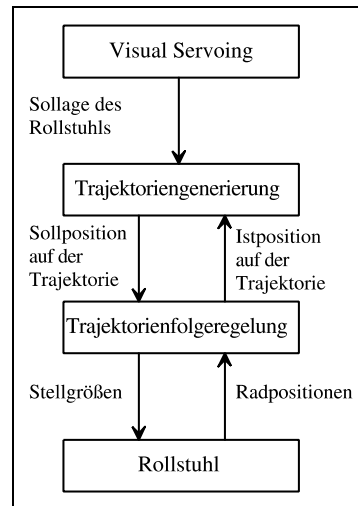


Bild 1: Struktur des Navigationssystems.

bezogen wird. Die Kombination aus Visual Servoing und Trajektorienfolgeregelung zur autonomen Navigation des Rollstuhls wird in Kapitel 4 erläutert. In einer Erweiterung des Verfahrens wird die CCD-Kamera drehbar auf einem Schwenk-Neige-Kopf montiert. Eine unterlagerte Landmarkenverfolgung sorgt dafür, dass die Landmarke während der Fahrt des Rollstuhls im Kamerabild verbleibt. Die einzelnen Verfahren werden am Beispiel eines Docking-Manövers am realen System überprüft.

2 Mobile Plattform und Fahrtregler

Bei der mobilen Plattform handelt es sich um einen motorisierten elektrischen Rollstuhl, der zur Unterstützung von älteren oder behinderten Menschen weiterentwickelt wird. Für dieses System sollen verschiedene visuelle Regler entworfen und ihre Leistungsfähigkeit am realen System untersucht werden. Ziel ist es, den Rollstuhl aufgrund von Informationen aus dem Kamerabild in eine gewünschte Lage relativ zu einem Objekt zu überführen (*Docking-Manöver*). Bild 2 zeigt den realen Systemaufbau mit montierter Entfernungsbildkamera.



Bild 2: Reales System.

Der Rollstuhl besitzt zwei unabhängig voneinander angetriebene Hinterräder zur Steuerung und zwei nichtangetriebene Vorderräder zur Stützung des Rollstuhls. Die beiden Elektromotoren an den Hinterrädern werden derart angesteuert, dass die Vorwärtsgeschwindigkeit u_1 und die Winkelgeschwindigkeit u_2 als Stellgrößen des Systems angesehen werden können. Sie ermöglichen Bewegungen des Rollstuhls in drei Freiheitsgraden auf ebenem Boden. Das System besitzt eine begrenzte kinematische Bewegungsfreiheit, da es sich nicht quer zur Fahrtrichtung bewegen kann (nichtholonome Eigenschaft). Das Modell des autonomen mobilen Rollstuhls wird nach [11] durch die Gln. (1)–(3) beschrieben, vgl. Bild 3. Das zugehörige Koordinatensystem ist durch das links hochgestellte Symbol gekennzeichnet.

$${}^W\dot{x} = u_1 \cdot \cos({}^W\Theta) \quad (1)$$

$${}^W\dot{y} = u_1 \cdot \sin({}^W\Theta) \quad (2)$$

$${}^W\dot{\Theta} = u_2 \quad (3)$$

Bild 3 zeigt den Zusammenhang zwischen dem Weltkoordinatensystem $\{W\}$ und dem Kamerakoordinatensystem $\{C\}$. Der Zustandsvektor des Rollstuhls ${}^W\underline{p} = [{}^Wx, {}^Wy, {}^W\Theta]^T$

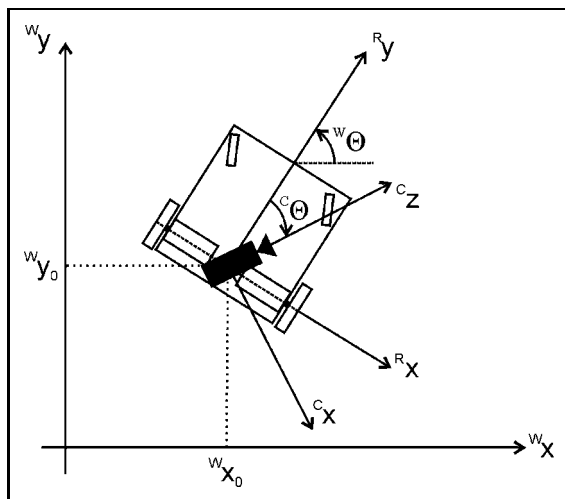


Bild 3: Rollstuhl-Kamera-Konfiguration (Draufsicht).

wird durch die beiden kartesischen Koordinaten Wx und Wy in der Bewegungsebene, sowie den Drehwinkel ${}^W\Theta$ beschrieben. Dieser Drehwinkel entspricht dem Winkel zwischen Fahrtrichtung des Rollstuhls und der x -Achse des Weltkoordinatensystems. Um die visuelle Regelung zu ermöglichen, wird eine Kamera am Rollstuhl befestigt. Die Kamera befindet sich direkt im Drehpunkt $({}^Wx_0, {}^Wy_0)$ des Rollstuhls.

Mithilfe der Kamera soll nun eine Ziel-Lage für den Rollstuhl ermittelt werden, um den Rollstuhl dann aus seiner aktuellen Start-Lage in diese Ziel-Lage fahren zu lassen. Dazu muss der Rollstuhl die Fähigkeit erhalten, einer vorgegebenen Trajektorie zu folgen. Die zu fahrende Trajektorie wird mittels einer übergeordneten Software generiert [11]. Diese Software erzeugt eine für den Rollstuhl fahrbare Trajektorie mit „weichem“ Verlauf zwischen der Start- und Ziel-Lage (Block „Trajektoriengenerierung“ in Bild 4). Die Bahn wird so berechnet, dass der Rollstuhl nicht nur die Zielkoordinaten Wx_z und Wy_z erreicht, sondern auch den vorgegebenen Zielwinkel ${}^W\Theta_z$. Weiterhin wurde eine Trajektorienfolgeregelung entworfen, die es ermöglicht, den Rollstuhl auf der vorgegebenen Trajektorie zu führen [11]. Dazu werden mit Drehgebern an den Antriebsrädern des Rollstuhls die Radpositionen zu äquidistanten Zeitpunkten ermittelt, sodass die aktuelle Lage des Rollstuhls jederzeit berechnet werden kann (Block „Odometrie“). Die Trajektorienfolgeregelung sorgt dafür, dass Anfangsfehler und Abweichungen von der vorgegebenen Trajektorie ausgeregelt werden und hat sich in zahlreichen Tests bewährt [11]. Im Entkopplungsregler wird ein Vergleich zwischen der Soll-Lage auf der Trajektorie und der Ist-Lage aus der Odometrie vorgenommen und es werden entsprechende Stellgrößen u_1 und u_2 generiert. Diese werden mittels unterlagerter PI-Regelungen schnell und stationär genau in Geschwindigkeiten des Rollstuhls umgesetzt (die Dynamik dieser Geschwindigkeitsregelungen wurde in Gln. (1)–(3) vernachlässigt). Bild 4 zeigt das zugehörige Blockschaltbild, in dem sich der Fahrtregler aus Trajektoriengenerierung und Trajektorienfolgeregelung zusammensetzt (vgl. Bild 1).

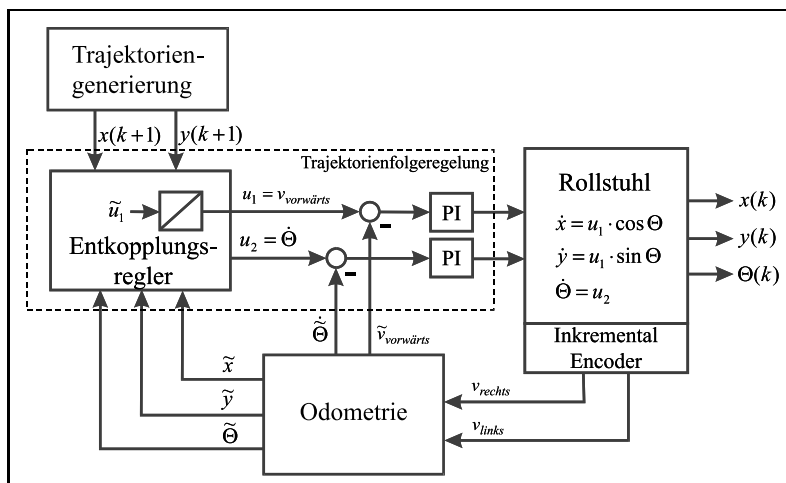


Bild 4: Blockschaltbild des Fahrtreglers.

3 Visual Servoing

Unter *Visual Servoing* versteht man die allgemeine Aufgabe, einen Roboter aufgrund von visuellen Informationen einer Kamera zu bewegen [12]. Dabei werden im geschlossenen Regelkreis fortlaufend Bilder aufgenommen, und der Roboter wird aufgrund der gewonnenen Bildinformationen bewegt, bis die gestellte Aufgabe erfüllt ist. Wird zusätzlich die Stellgröße für den Roboter allein und direkt aus Bildinformationen berechnet (d. h. ohne Berechnung der absoluten Lage im Weltkoordinatensystem), spricht man von einer so genannten *Image-Based Visual Servoing* (IBVS) Struktur, welche im Folgenden zur visuellen Regelung des Rollstuhls verwendet wird.

3.1 Einleitung

Ein Problem beim herkömmlichen Image-Based Visual Servoing resultiert daraus, dass die Entfernungen zwischen der Kamera und den aufgenommenen Bildmerkmalen (hier so genannten Blobs) unbekannt sind. In Kapitel 3.3 wird ein erweitertes Visual Servoing Verfahren vorgestellt, bei dem diese Entfernungen berücksichtigt werden und mit in die Regelung eingehen. Das Verfahren zur Ermittlung der Entfernungen ist von der jeweils verwendeten Kamera abhängig. Wird eine CCD-Kamera am Rollstuhl montiert, so müssen die Entfernungen aus dem Bild heraus geschätzt, also *indirekt* ermittelt werden. Dies geschieht über einen Vergleich der aktuellen Objektgröße im Bild mit einer Referenzgröße bei bekannter Kamera-Lage [13]. Wird hingegen eine spezielle Entfernungsbildkamera benutzt, so steht die Entfernungsinformation zu Objekten im Kamerabild *direkt* als Messgröße zur Verfügung. Zusätzlich zum Grauwertbild generiert diese Kamera ein Entfernungsbild, indem Laserblitze ausgesendet und von den aufgenommenen Objekten reflektiert werden. Die Laserblitze kommen nach ihrer individuellen Laufzeit, entsprechend ihren Entfernungen zum Objekt, wieder auf dem CCD-Chip der Kamera an [14]. Bei der Bestimmung der Entfernungen ist zu beachten, dass es sich um zwei verschiedene Entfernungswerte handelt: Bei der Schätzung mittels CCD-Kamera ergibt sich ein einziger Entfernungswert c_z für alle

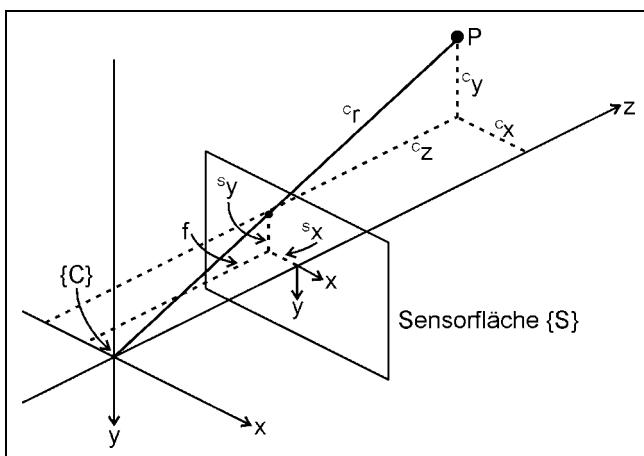


Bild 5: Kameramodell.

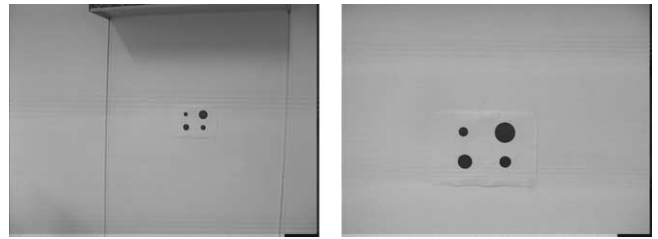


Bild 6: Kamerabilder der Landmarke (4 Blobs) in Start-Lage (links) und in Ziel-Lage (rechts).

vier Blobs, während die Laserkamera die Strecke c_r zu jedem einzelnen Blob ermittelt. Dieser Zusammenhang ist in Bild 5 dargestellt und bei der Anwendung des Visual Servoing zu berücksichtigen.

Bild 6 zeigt die verwendete Landmarke, bestehend aus vier schwarzen Kreisen (Blobs) auf weißem Hintergrund.

3.2 Herkömmliches 2D-Visual Servoing

Nur mit Hilfe der visuellen Informationen einer konventionellen CCD-Kamera soll ein Fahrmanöver des Rollstuhls in eine Lage relativ zu einer Landmarke erzeugt werden. Dazu werden dem Regler eindeutige Merkmale aus dem Kamerabild durch die Bildverarbeitung zur Verfügung gestellt. Diese Bildmerkmale sind hier die Mittelpunktkoordinaten von vier Blobs (siehe Bild 6), die durch eine Blobanalyse aus dem Kamerabild extrahiert werden. Die Mittelpunktkoordinaten s_{x_i} und s_{y_i} ($i = 1, \dots, 4$) werden im Sensorkoordinatensystem $\{S\}$ angegeben, welches sich auf den CCD-Chip der Kamera bezieht (siehe Bild 5). Für einen Blob zeigt Gl. (4) den Merkmalvektor, welcher abhängig von der Kamera-Lage ${}^c p$ ist:

$${}^s \underline{x} = {}^s \underline{x}({}^c p) = \begin{bmatrix} s_x \\ s_y \end{bmatrix} \quad (4)$$

Durch zeitliche Ableitung dieser Gleichung ergibt sich eine Beziehung zwischen der Ableitung des Merkmalvektors und dem Geschwindigkeitsvektor der Kamerabewegung:

$${}^s \dot{\underline{x}} = J({}^c p) \cdot {}^c \dot{p} \quad (5)$$

Die Matrix $J({}^c p)$ wird Bild-Jacobi-Matrix genannt und liefert einen Zusammenhang zwischen einer Änderung der Kamera-Lage und der entsprechenden Änderung der Lage der Bildmerkmale:

$$J({}^c p) = \frac{\partial {}^s \underline{x}}{\partial {}^c p} \quad (6)$$

Aus einem Vergleich zwischen momentanen und gewünschten Merkmalen (Bild 6) berechnet der Regler eine notwendige Änderung der Kamera-Lage, um den Bildfehler zu minimieren. Diese neue Kamera-Lage lässt sich durch Koordinatentransformation in die neue Rollstuhl-Lage (Weltkoordinatensystem) umrechnen. Die gewünschten Merkmale ${}^s \underline{x}_{soll}$ in der Ziel-Lage müssen vorab definiert werden, was durch „erlernen“ (*teaching-by-showing*, [2]) geschieht. Dazu wird der Rollstuhl in die Ziel-Lage (z. B. eine relative Lage zu einem Tisch) gefahren, das zugehörige Bild

aufgenommen, die Merkmale extrahiert und als Sollwerte gespeichert. Während der Regelung werden die momentanen Bildmerkmale in die Sollbildmerkmale überführt. Ist dies geschehen, befindet sich der Rollstuhl in derselben relativen Lage zum Objekt wie beim Lernvorgang. Um eine Bewegung der Kamera berechnen zu können, hat der Regler die Aufgabe, aus einer Regelabweichung im zweidimensionalen Sensorkoordinatensystem $\{S\}$ Bewegungen im Kamerakoordinatensystem $\{C\}$ zu erzeugen. Diese Transformation wird mithilfe der Bild-Jacobi-Matrix J durchgeführt, die bereits in Gl. (6) eingeführt wurde und nun in Gl. (7) detailliert gezeigt ist. Diese Matrix stellt eine lineare Transformation von einer Änderung der Kamera-Lage $\underline{c}_{\dot{p}}$ (Ursache), beschrieben in Kamerakoordinaten, in eine entsprechende Änderung der Projektion eines ruhenden Punktes $\underline{s}_{\dot{x}}$ (Wirkung), beschrieben in Sensorkoordinaten, dar [2].

$$\underbrace{\begin{bmatrix} s_{\dot{x}} \\ s_{\dot{y}} \\ s_{\dot{z}} \end{bmatrix}}_{\underline{s}_{\dot{x}}} = \underbrace{\begin{bmatrix} -\frac{f}{c_z} & 0 & \frac{s_x}{c_z} & \frac{s_x s_y}{f} & -\frac{f^2 + s_x^2}{f} & s_y \\ 0 & -\frac{f}{c_z} & \frac{s_y}{c_z} & \frac{f^2 + s_y^2}{f} & -\frac{s_x s_y}{f} & -s_x \end{bmatrix}}_J \cdot \underbrace{\begin{bmatrix} t_x \\ t_y \\ t_z \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}}_{\underline{c}_{\dot{p}}} \quad (7)$$

Gl. (7) zeigt den Zusammenhang für einen einzelnen Blob (zwei Merkmale) und eine mögliche Kamerabewegung in sechs Freiheitsgraden (dabei repräsentieren t_i und ω_i die translatorischen bzw. rotatorischen Geschwindigkeiten). Pro Blob wächst die Bild-Jacobi-Matrix um zwei Zeilen an. Wird die Kamera nicht in allen sechs Freiheitsgraden bewegt, fallen die entsprechenden Spalten der Bild-Jacobi-Matrix weg. Im vorliegenden Fall der am Rollstuhl montierten Kamera gilt $t_y = \omega_x = \omega_z = 0$, und der die Änderung der Kamera-Lage beschreibende Vektor ergibt sich zu:

$$\underline{c}_{\dot{p}} = \begin{bmatrix} t_x \\ t_z \\ \omega_y \end{bmatrix} \quad (8)$$

Die Bild-Jacobi-Matrix J ist eine Funktion der momentanen Brennweite f der Kamera, der aktuellen Sensorparameter s_x und s_y , sowie der momentanen Entfernung c_z zwischen dem Blob und der x -Achse des Kamerakoordinatensystems $\{C\}$, siehe Bild 5. Beim herkömmlichen Visual Servoing besteht das Problem, dass c_z unbekannt ist. Es ist möglich, dieses Problem zu umgehen, indem eine *konstante* Bild-Jacobi-Matrix verwendet wird, welche die Werte aus der Ziel-Lage benutzt, was zur Folge hat, dass der Visual Servoing Algorithmus nur in einem relativ kleinen Bereich um die Ziel-Lage herum zufriedenstellend arbeitet [2]. Befindet sich die Start-Lage weit von der Ziel-Lage entfernt, ist es möglich, dass die Bildmerkmale während der visuellen Regelung aus dem Kamerabild herauslaufen und es so zum Versagen der Regelung kommt.

Alternativ kann mit einer *adaptiven* Bild-Jacobi-Matrix gearbeitet werden, in der die Entfernung c_z zwischen Kamera

und Blobs regelmäßig geschätzt bzw. gemessen wird, so dass in der Bild-Jacobi-Matrix stets mit den aktuellen Blobskoordinaten (s_{x_i} , s_{y_i} und c_{z_i}) gerechnet werden kann. Wird der Abstand c_z als weitere Messgröße aufgefasst, so liegt es nahe, die Bild-Jacobi-Matrix um eine Zeile pro Blob zu *erweitern* und so einen Zusammenhang zwischen der Änderung der Kamera-Lage und der entsprechenden Entfernungsänderung herzustellen. Wie im Folgenden beschrieben wird, hat dies unmittelbar zur Folge, dass die Entfernung zu den Blobs als weitere Regelgröße in den Algorithmus eingeht.

3.3 3D-Visual Servoing

Beim 3D-Visual Servoing wird die Entfernung zwischen Kamera und Blobs direkt oder indirekt ermittelt und deshalb als drittes Element dem Merkmalvektor hinzugefügt. Zu beachten ist, dass es sich, je nach verwendeter Kamera, um verschieden definierte Entfernungen handelt (vgl. Bild 5). Gl. (9) zeigt den erweiterten Merkmalvektor für einen Blob jeweils für die CCD-Kamera und die Entfernungsbildkamera (EBK):

$$\underline{s}_{\dot{x}_{CCD}} = \begin{bmatrix} s_x \\ s_y \\ c_z \end{bmatrix} \quad \text{bzw.} \quad \underline{s}_{\dot{x}_{EBK}} = \begin{bmatrix} s_x \\ s_y \\ c_r \end{bmatrix} \quad (9)$$

Durch das hinzugekommene Merkmal wird auch die Bild-Jacobi-Matrix um eine Zeile erweitert. Gl. (10) zeigt die resultierende Matrix für einen Blob und die drei Freiheitsgrade der Kamera bzw. des Rollstuhls bei Verwendung der CCD-Kamera:

$$\underbrace{\begin{bmatrix} s_{\dot{x}} \\ s_{\dot{y}} \\ c_{\dot{z}} \end{bmatrix}}_{\underline{s}_{\dot{x}_{CCD}}} = \underbrace{\begin{bmatrix} -\frac{f}{c_z} & \frac{s_x}{c_z} & -\frac{f^2 - s_x^2}{f} \\ 0 & \frac{s_y}{c_z} & -\frac{s_x s_y}{f} \\ 0 & -1 & \frac{s_x c_z}{f} \end{bmatrix}}_{J_{CCD}} \cdot \underbrace{\begin{bmatrix} t_x \\ t_z \\ \omega_y \end{bmatrix}}_{\underline{c}_{\dot{p}}} \quad (10)$$

Wird die Entfernungsbildkamera verwendet, muss berücksichtigt werden, dass nicht wie bisher c_z , sondern die Entfernung c_r zu den Blobs gemessen wird. Die resultierende Bild-Jacobi-Matrix zeigt Gl. (11):

$$\underbrace{\begin{bmatrix} s_{\dot{x}} \\ s_{\dot{y}} \\ c_{\dot{r}} \end{bmatrix}}_{\underline{s}_{\dot{x}_{EBK}}} = \underbrace{\begin{bmatrix} -\frac{\sqrt{s_x^2 + s_y^2 + f^2}}{c_r} & \frac{s_x \sqrt{s_x^2 + s_y^2 + f^2}}{f c_r} & -\frac{f^2 - s_x^2}{f} \\ 0 & \frac{s_y \sqrt{s_x^2 + s_y^2 + f^2}}{f c_r} & -\frac{s_x s_y}{f} \\ -\frac{s_x}{\sqrt{s_x^2 + s_y^2 + f^2}} & -\frac{f}{\sqrt{s_x^2 + s_y^2 + f^2}} & 0 \end{bmatrix}}_{J_{EBK}} \cdot \underbrace{\begin{bmatrix} t_x \\ t_z \\ \omega_y \end{bmatrix}}_{\underline{c}_{\dot{p}}} \quad (11)$$

Mit den hier beschriebenen Erweiterungen des Visual Servoing, den in Kapitel 3.2 vorgestellten grundlegenden Prinzipien, sowie dem im Folgenden hergeleiteten Regelgesetz wird das Docking-Manöver für den Rollstuhl durchgeführt. Durch die Erweiterungen des 3D-Visual Servoing wird die Entfernungsinformation nicht nur zum Aktualisieren der Bild-Jacobi-Matrix benutzt, sondern geht erstmals beim IBVS auch als Regelgröße ein. Die Ergebnisse der Versuche am realen System sind in Kapitel 4 dargestellt.

3.4 Regelgesetz

Um robust gegenüber Messrauschen zu sein, sollte die Anzahl der Bildmerkmale größer gewählt werden als die Anzahl der Freiheitsgrade der Kamera. Hier wurden vier Blobs gewählt (siehe Bild 6). In diesem Fall ergibt sich aus Gl. (5) durch Bildung der Pseudoinversen J^+ folgende Näherungslösung für den gesuchten Vektor ${}^C \underline{\dot{p}}$:

$${}^C \underline{\dot{p}} = J^+ \cdot {}^S \underline{\dot{x}} \quad (12)$$

Die Komponenten der Stellgröße ${}^C \underline{\dot{p}}$ und des Merkmalvektors ${}^S \underline{\dot{x}}$ in Gl. (12) sind Geschwindigkeiten. Da es sich bei dem zu implementierendem Regelkreis um einen diskreten Abtastregelkreis handelt, ergibt sich ein Stellgesetz aus Gl. (12) unter Berücksichtigung einer Abtastzeit T_a . Die Aufgabe ist dann, folgende Fehlerfunktion zu minimieren:

$$\underline{e} = \Delta {}^S \underline{x} = {}^S \underline{x} - {}^S \underline{x}_{soll} \quad (13)$$

Ein geeignetes Verhalten des Regelfehlers $\underline{e}(t)$ lässt sich durch Verwendung einer Differentialgleichung 1. Ordnung (14) erreichen:

$$\dot{\underline{e}} = -K_f \cdot \underline{e} \quad \text{mit:} \quad \dot{\underline{e}} = \dot{{}^S \underline{x}} \quad (14)$$

Durch angemessene Wahl der diagonalen Verstärkungsmatrix K_f wird ein exponentieller Verlauf des Regelfehlers gegen null erreicht für $t \rightarrow \infty$. Durch Einsetzen von Gl. (13) in Gl. (14) und des Ergebnisses in Gl. (12), ergibt sich mit ${}^C \underline{\dot{p}} \approx \Delta {}^C \underline{p} / T_a$ das folgende zeitdiskrete Regelgesetz:

$$\Delta {}^C \underline{p} = -J^+ \cdot K_f \cdot T_a \cdot \Delta {}^S \underline{x} \quad (15)$$

Dieses Regelgesetz (15) ist sowohl für den Ansatz des herkömmlichen 2D-Visual Servoing, als auch für das 3D-Visual Servoing geeignet. Bild 7 zeigt den resultierenden Regelkreis, wobei der Block „visueller Regler“ das Stellgesetz (15) enthält. Der visuelle Regler ermittelt aus der Position der Merkmale im Bild eine neue Ziel-Lage für die Kamera $\Delta {}^C \underline{p} = [\Delta {}^C x, \Delta {}^C z, \Delta {}^C \Theta]^T$, die durch Koordinatentransformation in eine neue Ziel-Lage für den Rollstuhl ${}^W \underline{p}_{neu}$ umgerechnet werden kann. Diese neue Rollstuhl-Lage wird an den Fahrtregler übergeben, der die Stellgrößen für die Antriebsräder des Rollstuhls ermittelt.

Der bisher präsentierte Regelalgorithmus erweist sich als geeignet, solange der Rollstuhl seiner Ziel-Lage nicht zu nahe kommt. Um Probleme zu umgehen, die nahe der Ziel-

Lage auftreten, werden im Folgenden einige Erweiterungen dieses Regelalgorithmus vorgestellt.

4 Docking-Manöver mittels Visual Servoing

Zu Beginn des Docking-Manövers wird der Rollstuhl in eine Start-Lage gebracht, aus der die Merkmale für die Kamera sichtbar sind. In dieser Start-Lage wird nun vom visuellen Regler eine Ziel-Lage für die Kamera berechnet und vom Fahrtregler in Stellgrößen für den Rollstuhl umgesetzt. Da sich die Start-Lage relativ weit entfernt von der exakten Ziel-Lage befindet, ist die im ersten Schritt berechnete Ziel-Lage relativ ungenau. Deswegen werden während der Fahrt ständig neue Bilder aufgenommen und neue Ziel-Lagen berechnet. Je näher der Rollstuhl der Ziel-Lage kommt, desto genauer kann der visuelle Regler die exakte Ziel-Lage berechnen. Es wäre nun möglich, jeweils zwischen der aktuellen Lage des Rollstuhls und der berechneten Ziel-Lage eine Trajektorie zu planen. In den meisten real vorkommenden Fällen nimmt die Trajektorie jedoch einen s-förmigen Verlauf an (kubische Funktionen, [11]). Je näher nun der Rollstuhl seiner exakten Ziel-Lage kommt, desto kürzer werden die s-förmigen Trajektorien. Diese sind dann für den Rollstuhl nicht mehr fahrbar bzw. bieten dem Insassen keine „weiche“ Fahrt mehr. Deswegen wird hier vorgeschlagen, die während der Fahrt geplanten Trajektorien nicht von der aktuellen Lage des Rollstuhls, sondern von seiner ursprünglichen Start-Lage (zu Beginn des Docking-Manövers) aus zu planen. Dazu werden die Elemente von K_f aus Gl. (15) so groß gewählt, dass der visuelle Regler Ziel-Lagen in der Nähe der wirklichen Ziel-Lage berechnen kann. Die Berechnung von Trajektorien zwischen Start-Lage und den jeweils neu berechneten Ziel-Lagen wird dann durch eine Koordinatentransformation möglich, die in Bild 8 gezeigt ist.

In Bild 8 entspricht der Ursprung von $\{W\}$ der Start-Lage des Rollstuhls. Hat sich der Rollstuhl nun bereits in seine aktuelle Lage $\{R\}$ bewegt, entspricht dies der Transformation ${}^W_R H$, welche aus der Odometrie abgelesen werden kann. Die Kamera $\{C\}$ sei zunächst unbeweglich am Rollstuhl montiert und „blickt“ in Fahrtrichtung. Dadurch ist der Zusammenhang ${}^R_C H$ bekannt. Der visuelle Regler liefert die Stellgröße $\Delta {}^C \underline{p} = {}^C_C H$, welche die neue Kamera-Lage in Bezug auf die alte Kamera-Lage beschreibt. Die

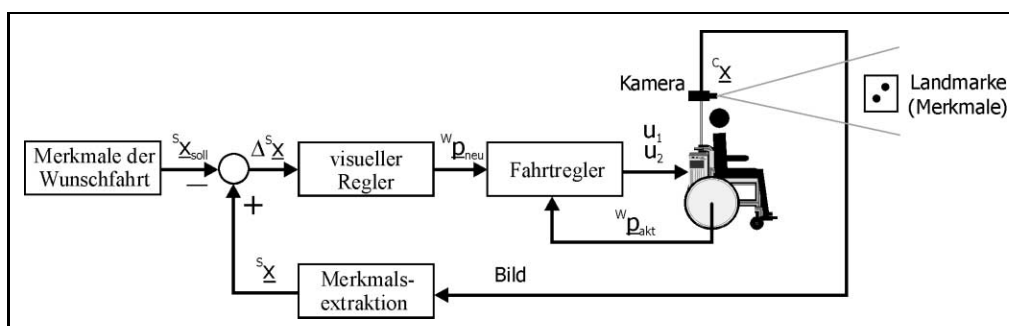


Bild 7: Visual Servoing des Rollstuhls.

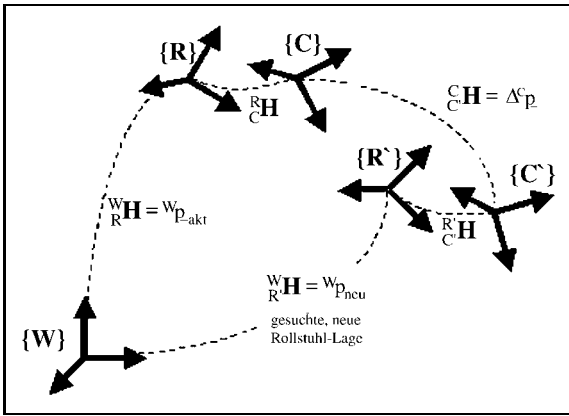


Bild 8: Berechnung der neuen Ziel-Lagen durch Koordinatentransformation.

neue Rollstuhl-Lage ist mit dieser neuen Kamera-Lage über ${}^R_C H = {}^R C H$ verbunden. Nun sind alle Transformationen bekannt und die neue Rollstuhl-Lage in $\{W\}$ kann über Gl. (16) berechnet werden:

$${}^W_R H = {}^W_R H \cdot {}^R_C H \cdot {}^C_C H \cdot {}^R_C H \quad (16)$$

4.1 Experiment mit CCD-Kamera

Im Verlauf der Fahrt des Rollstuhls können mithilfe von Gl. (16) ständig neue Trajektorien zwischen der ursprünglichen Start-Lage und den durch Visual Servoing neu berechneten Ziel-Lagen generiert werden. Es entsteht eine Trajektorienschar, wie im Folgenden genauer erläutert wird:

Während der Rollstuhlbewegung werden die Sollgrößen des jeweils aktuellen Trajektorienabschnitts an den Trajektorienfolgeregler übergeben, der den Rollstuhl auf der momentanen Trajektorie hält. Wird eine neue Ziel-Lage vom visuellen Regler berechnet, wird die aktuelle Trajektorie korrigiert [15]. In Bild 9 ist gut zu erkennen, wie

sich die Ziel-Lagen der einzelnen Trajektorien kontinuierlich von rechts nach links in Richtung der wirklichen Ziel-Lage bewegen. Dies wurde durch die Einführung des 3D-Visual Servoing möglich und ist vor allem der sehr guten Schätzung der Entfernung c_z und der damit verbundenen ähnlichen Form und Länge der einzelnen Trajektorien zu verdanken. Zusätzliche Fahrversuche mit 2D-Visual Servoing zeigten eine sprunghafte Entwicklung der Ziel-Lagen, sodass hier die Vorteile des 3D-Visual Servoing deutlich werden [16]. Die Trajektorienfolgeregelung führt den Rollstuhl auf die korrigierte Trajektorie. Während dieses Verfahrens werden alle gerade aktuellen Trajektorienabschnitte zusammengefügt und es entsteht eine resultierende Trajektorie, die den Rollstuhl in seine Ziel-Lage führt. Bei dem in Bild 9 dargestellten Versuch (der Drehwinkel ist hier nicht eingezeichnet) erreichte der Rollstuhl die Ziel-Lage ${}^W \underline{p} = [25 \text{ mm}, 2819 \text{ mm}, 68^\circ]$. Die wirkliche Ziel-Lage, in der die Merkmale angelernt wurden, lag bei ${}^W \underline{p}_{soll} = [-15 \text{ mm}, 2809 \text{ mm}, 67^\circ]$. Die Abweichungen erklären sich hauptsächlich durch die nicht exakte Berechnung der Ziel-Lage durch den visuellen Regler. Zusätzliche Fahrexperimente zeigten maximale Abweichungen in ähnlichen Größenordnungen [17]. Bedenkt man die geringe Dynamik des Systems Rollstuhl und die Zielsetzung der zu lösenden Aufgabe, kann man jedoch von einem guten Regelergebnis sprechen.

4.2 Experiment mit Entfernungsbildkamera

Das in Bild 10 dargestellte Experiment zeigt die Umsetzung des Docking-Manövers mithilfe der Entfernungsbildkamera. Der Rollstuhl erreichte seine Ziel-Lage bei ${}^W \underline{p} = [100 \text{ mm}, 2131 \text{ mm}, 119^\circ]$, während die wirkliche Ziel-Lage bei ${}^W \underline{p} = [200 \text{ mm}, 2200 \text{ mm}, 100^\circ]$ lag. Der größere Fehler im Vergleich zur Verwendung der CCD-Kamera entsteht, da wegen der hohen Abtastzeiten (ca.

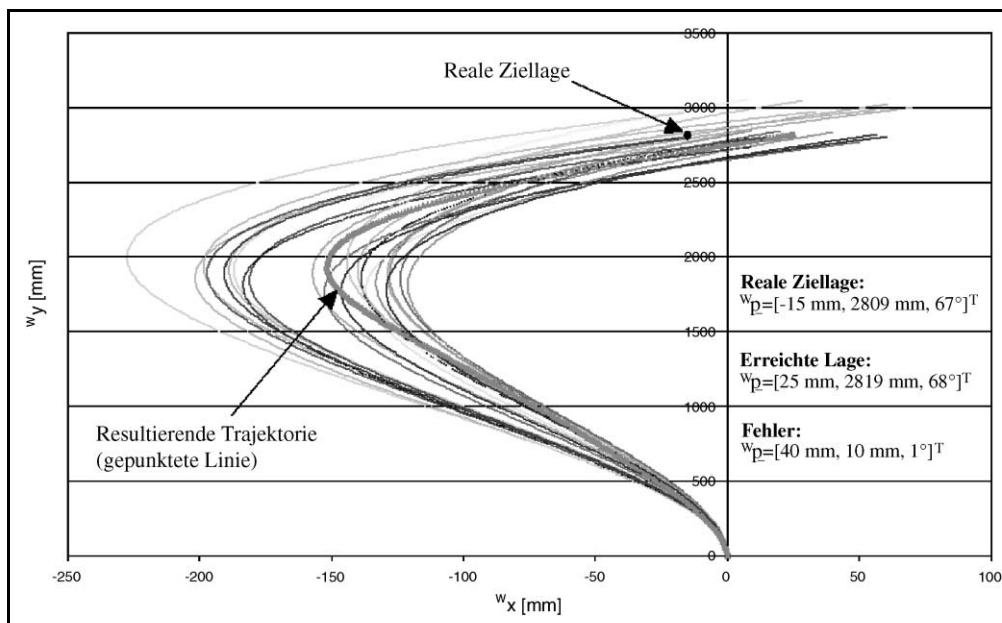


Bild 9: Trajektorienschar und resultierende Trajektorie bei Verwendung der CCD-Kamera und 3D-Visual Servoing.

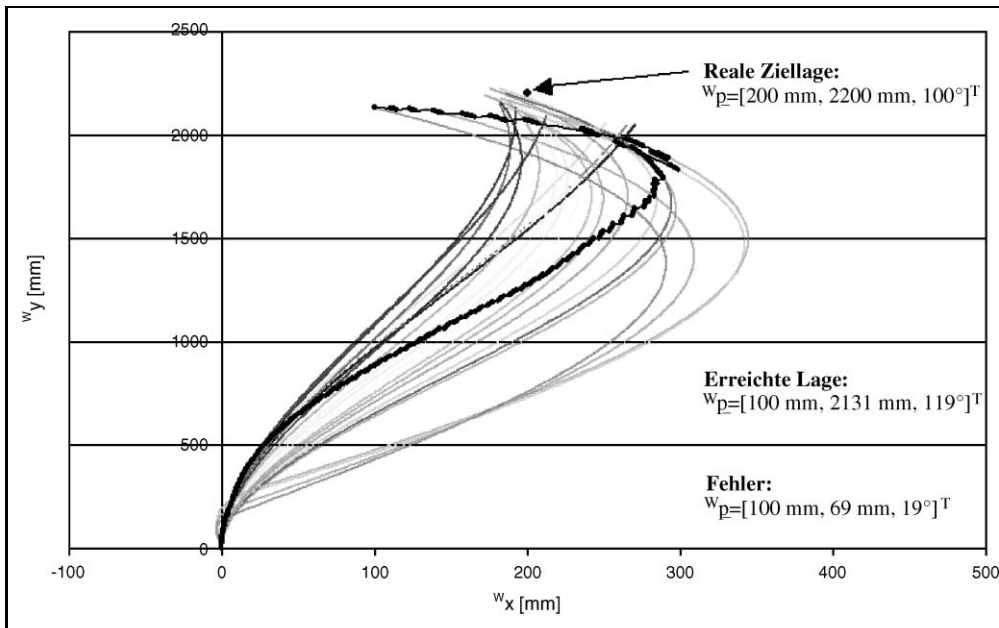


Bild 10: Trajektorienschär und resultierende Trajektorie bei Verwendung der Entfernungsbildkamera.

350 ms) nicht genug Trajektorien erzeugt werden können. Diese hohen Abtastzeiten werden durch die Bildaufnahme und Bildverarbeitung verursacht, da die Entfernungsbilder stark verrauscht sind. Deswegen müssen zusätzliche Filter-Algorithmen angewendet werden, um die korrekten Entfernungswerte zu bestimmen. Während dieser Zeit fährt der Rollstuhl weiter, sodass nicht genug Ziel-Lagen vom visuellen Regler berechnet werden können. Trotz der zusätzlichen Filter verändern sich die gemessenen Entfernungen stark von einer Bildaufnahme zur nächsten, sodass der visuelle Regler keine Ziel-Lagen berechnen kann, die sich (wie im Falle der CCD-Kamera) kontinuierlich in Richtung der wirklichen Ziel-Lage bewegen. Erschwerend kommt hinzu, dass die berechneten Trajektorien in ihrer Form und Länge variieren, was zu Sprüngen in der resultierenden Trajektorie führt (siehe Bild 10). Daraus resultieren Fehler in der Trajektorienfolgeregelung, was zu einem schlechteren Gesamtergebnis als bei Verwendung der CCD-Kamera führt. Für die Zukunft ist allerdings zu erwarten, dass Verbesserungen der Lasertechnik und Steigerungen der Rechnerleistungen bessere Regelergebnisse ermöglichen werden.

4.3 Experiment mit drehbarer CCD-Kamera

Um das Visual Servoing erfolgreich durchführen zu können, ist es notwendig, die Bildmerkmale während des gesamten Regelvorgangs im Kamerabild zu halten. Für den in Kapitel 4.1 und 4.2 dargestellten Fall einer unbeweglich am Rollstuhl fixierten Kamera resultiert die obige Bedingung in einem relativ kleinen Arbeitsbereich für den Rollstuhl. Um dieses Problem zu umgehen und den Arbeitsbereich zu vergrößern, wird die CCD-Kamera nun auf einem Schwenk-Neige-Kopf drehbar am Rollstuhl befestigt.

Zur Regelung des Drehwinkels des Schwenk-Neige-Kopfs wird eine eigenständige Landmarkenverfolgung in den Block „visueller Regler“ in Bild 7 integriert. Diese

Landmarkenverfolgung aktualisiert den Drehwinkel des Schwenk-Neige-Kopfs und hält so die Landmarke in der Bildmitte. Dazu werden die Koordinaten der Merkmale auf dem Sensorchip $\{S\}$ ermittelt und geeignete Stellgrößen zur Drehung des Schwenk-Neige-Kopfs generiert. Durch die Drehung der Kamera entsteht ein Differenzwinkel ${}^C\Theta$ zwischen der Orientierung von Kamera und Rollstuhl (vgl. Bild 3). Dieser Winkel ist *nicht* in dem Regelgesetz (15) berücksichtigt, da dort davon ausgegangen wird, dass Kamera und Rollstuhl die gleiche Orientierung besitzen. Um die in Kapitel 3 beschriebenen Aussagen über das Visual Servoing beibehalten zu können, wird im Folgenden die Methode der *virtuellen Sensorebene* neu eingeführt:

Bild 11a) zeigt das Kamerakoordinatensystem $\{C\}$ für den Fall, dass die Landmarkenverfolgung die Bildmerkmale in die Mitte des Kamerabildes gebracht hat. Es sei angenommen, dass in diesem Fall Kamera und Rollstuhl *nicht* die gleiche Orientierung besitzen. Die Bildmerkmale werden auf der physikalischen Sensorebene abgebildet.

Nun wird das Kamerakoordinatensystem $\{C\}$ in das System $\{C'\}$ transformiert, welches die gleiche Orientierung wie das Rollstuhlkoordinatensystem $\{R\}$ besitzt. Diese Transformation kann durchgeführt werden, weil der Differenzwinkel ${}^C\Theta$ bekannt ist. Mithilfe der Gleichungen des Lochkameramodells können nun die Bildmerkmale berechnet werden, wie sie auf der mitgedrehten Sensorebene $\{S'\}$ erscheinen würden. Wie in Bild 11b) gezeigt ist, müssen die resultierenden Koordinaten der Bildmerkmale nicht auf der physikalisch vorhandenen Sensorebene (dem CCD-Chip) liegen. Statt dessen können sie auf einer virtuellen Sensorebene angeordnet sein, deren Ausmaße theoretisch unendlich sind.

Mithilfe des oben beschriebenen Verfahrens der virtuellen Sensorebene kann das in Kapitel 3 beschriebene Visual Servoing ohne Modifikationen angewendet werden, auch wenn die Kamera gegenüber dem Rollstuhl ver-

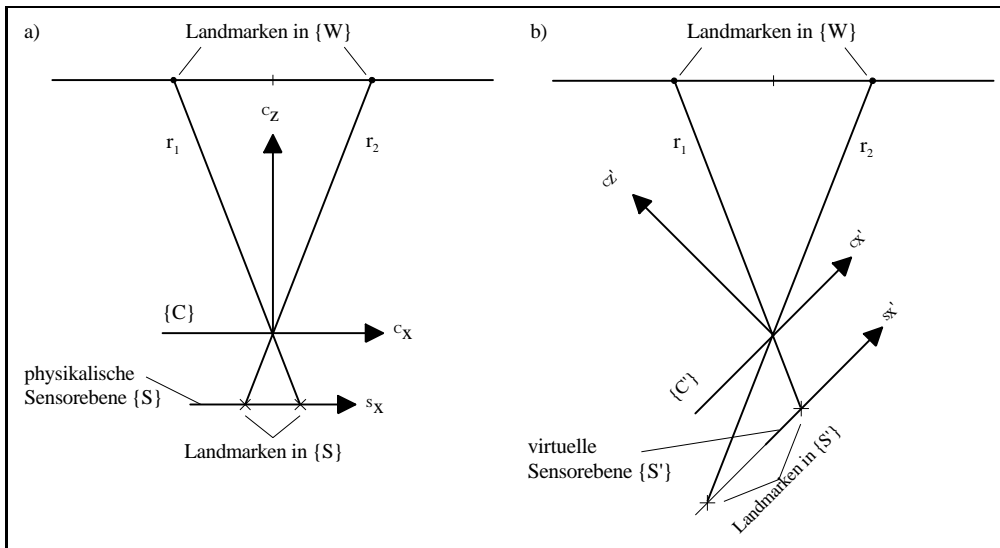


Bild 11: a) Ungedrehtes System (Lochkameramodell, Draufsicht) und b) Gedrehtes System (virtuelle Sensorebene).

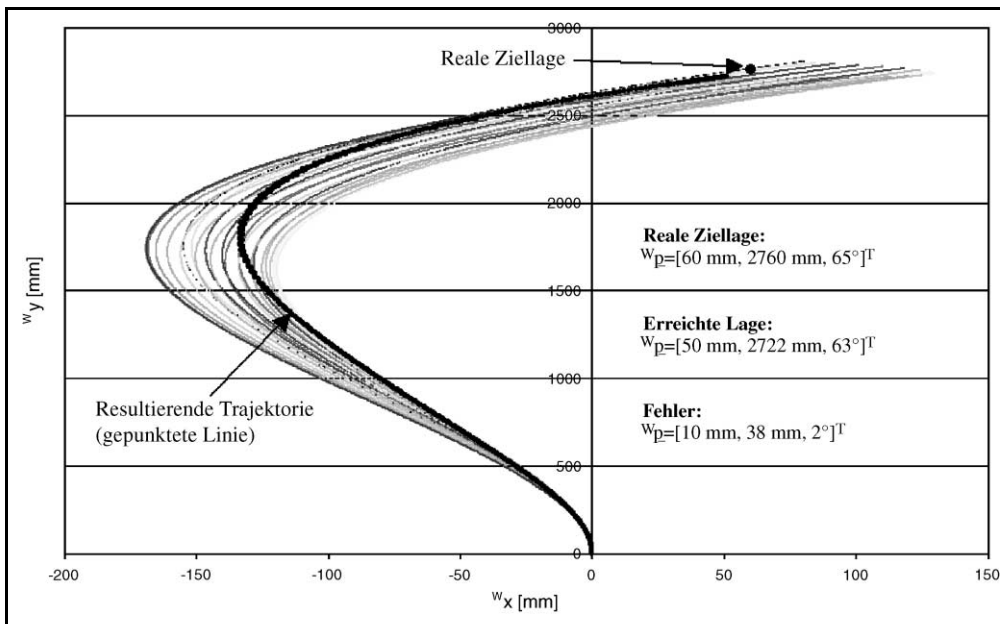


Bild 12: Trajektorienschar und resultierende Trajektorie bei Verwendung der drehbar gelagerten CCD-Kamera.

dreht ist. Der von der Landmarkenverfolgung verursachte Differenzwinkel ${}^C\Theta$ wird durch die Transformation der Bildmerkmale auf die virtuelle Sensorebene berücksichtigt. Bild 12 zeigt ein entsprechendes Experiment. Es ist wiederum ein Docking-Manöver gezeigt, bei dem diesmal eine Start-Lage gewählt wird, aus der die Merkmale bei fest in Fahrtrichtung montierter Kamera nicht sichtbar wären. Der Schwenk-Neige-Kopf wird jedoch von der Landmarkenverfolgung so gedreht, dass die Merkmale während der gesamten Fahrt im Kamerabild zu sehen sind. Der dargestellte Versuch zeigt sehr dicht beieinanderliegende Trajektorien. Durch die Verwendung des 3D-Visual Servoing wurde erreicht, dass alle während der Fahrt berechneten Ziel-Lagen kontinuierlich um die reale Ziel-Lage herum angeordnet sind. Das Ergebnis weist große Ähnlichkeiten mit dem in Kapitel 4.1 dargestellten Versuch ohne Landmarkenverfolgung auf und liefert mit einem nur geringen Fehler

${}^W p_{Fehler} = [10 \text{ mm}, 38 \text{ mm}, 2^\circ]^T$ ein sehr gutes Ergebnis des Docking-Manövers.

5 Zusammenfassung

Es wurde beschrieben, wie mithilfe des Image-Based Visual Servoing ein Docking-Manöver einer mobilen Plattform durchgeführt werden kann. Der Ansatz des Image-Based Visual Servoing wurde erweitert, indem die geschätzte bzw. gemessene Entfernung zu den Bildmerkmalen in den Merkmalvektor und in das Regelgesetz eingefügt wurde. Ein entsprechendes zeitdiskretes Regelgesetz, das die erweiterte Bild-Jacobi-Matrix benutzt, wurde hergeleitet. Eine zusätzliche Landmarkenverfolgung wurde in das Visual Servoing integriert, sodass die Landmarke nicht aus dem Kamerabild herausläuft. Durch die Verwendung einer unterlagerten Tra-

jektorienfolgeregelung stellen die präsentierten Verfahren eine geeignete Methode zur Navigation einer nichtholonomen mobilen Plattform dar.

Literatur

- [1] *Kolmanovsky, I., McClamroch, N. H.*: Developments in Non-holonomic Control Problems, IEEE Control Systems Magazine, vol. 15, no. 6, S. 20–36, Dezember 1995.
- [2] *Hutchinson, S., Hager, G., Corke, P.*: A Tutorial on Visual Servo Control, IEEE Trans. on Robotics and Automation vol. 12, no. 5, S. 651–670, 1996.
- [3] *Kämpke, T., Strobel, M.*: Navigation nicht-kreisförmiger mobiler Roboter in hindernisdichten Umgebungen, In Wörn, H., Dillmann, R., Henrich, D. (Hrsg.), Informatik aktuell, Autonome Mobile Systeme 1998, S. 156–163, Springer Verlag, 1998.
- [4] *Tarin, C., Brugger, H., Tibken, B., Hofer, E. P.*: Globally Asymptotically Stable Position Control for an Autonomous Mobile Robot, Proceedings of European Control Conference (ECC '99), Germany, 1999.
- [5] *Wong, A. K. C., Gao, S.*: Vision Directed Path Planning, Navigation and Control for an Autonomous Mobile Robot, Proceedings of the SPIE – The International Society for Optical Engineering, vol. 1831, S. 352–360, 1992.
- [6] *Buhmann, J., Burgard, W., Cremers, A. B., Fox, D., Hofmann, T., Schneider, F., Strikos, J., Thrun, S.*: The Mobile Robot Rhino, AI Magazine, vol. 16, no. 2, S. 31–38, 1995.
- [7] *Dulimarta, H. S., Jain, A. K.*: Mobile Robot Localization in Indoor Environment, Pattern Recognition, vol. 30, no. 1, S. 99–111, 1997.
- [8] *Rous, M., Matsikis, A., Broicher, F., Kraiss, K.-F.*: Optische Navigation für den assistierten Betrieb eines Rollstuhls, In Levi, P., Schanz, M., (Hrsg.), Informatik aktuell, Autonome Mobile Systeme 2001, S. 29–35, Springer Verlag, 2001.
- [9] *Masutani, Y., Mikawa, M., Maru, N., Miyazaki, F.*: Visual Servoing for Non-Holonomic Mobile Robots, IROS '94. Proc. of the IEEE/RSJ/GI Int. Conf. on Intelligent Robots and Systems, S. 1133–40 vol. 2, 1994.
- [10] *Pissard-Gibollet, R., Rives, P.*: Applying Visual Servoing Techniques to Control a Mobile Hand-Eye System, Proc. of IEEE Int. Conf. on Robotics and Automation, S. 166–171, 1995.
- [11] *Buttelmann, M., Lohmann, B., Kieren, M.*: Trajektoriengenerierung und Bahnregelung für nichtholonome, autonome Fahrzeuge, In Schmidt, G., Hanebeck, U., Freyberger, F. (Hrsg.), Informatik aktuell, Autonome Mobile Systeme 1999, S. 303–312, Springer Verlag 1999.
- [12] *Corke, I.*: Visual Control of Robots: High-Performance Visual Servoing, Research Studies Press Ltd. oder John Wiley & Son Inc., 1997.
- [13] *Lang, O.*: Bildbasierte Roboterregelung mit einer am Greifer montierten Zoomkamera, Fortschritt-Berichte VDI, Reihe 8, Nr. 857, VDI Verlag 2000.
- [14] *Forgber, E.*: High Resolution 3D-Camera, Daimler-Benz Aerospace, Space Infrastructure Division, New Products & Avionics Software Dept., 1998.
- [15] *Lietmann, T.*: Visuelle Regelung zur autonomen Navigation einer nichtholonomen mobilen Plattform, Fortschritt-Berichte VDI, Reihe 8, Nr. 933, VDI Verlag 2002.
- [16] *Lietmann, T., Lohmann, B.*: 3D-Visual Servoing zur autonomen Navigation einer nichtholonomen mobilen Plattform, In Levi, P., Schanz, M. (Hrsg.), Informatik aktuell, Autonome Mobile Systeme 2001, S. 47–54, Springer Verlag, 2001.
- [17] *Lietmann, T., Lohmann, B.*: Docking-Procedure for a Non-Holonomic Mobile Platform Based on Visual Servoing, Proceedings of 6th International IFAC Symposium on Robot Control, SYROCO 2000, Vienna, 2000.

Manuskripteingang: 25. Februar 2002.



Dr.-Ing. Thorsten Lietmann war bis Februar 2002 wissenschaftlicher Mitarbeiter am Institut für Automatisierungstechnik der Universität Bremen. Dort beschäftigte er sich im Fachgebiet „Systemdynamik und Regelungstechnik“ mit der Entwicklung von neuen Verfahren zur autonomen Navigation von mobilen Plattformen. Seit März 2002 arbeitet er als Entwickler für System-Software bei der Firma benntec Systemtechnik GmbH in Bremen. Hauptarbeitsgebiete: Bildverarbeitungssysteme, Software-Entwicklung.

Adresse: benntec Systemtechnik GmbH, Abteilung Bildverarbeitungs- und Inspektionssysteme, Walter-Geerdes-Str. 10–12, D-28307 Bremen, E-Mail: t.lietmann@benntec.de, www.benntec.de



Prof. Dr.-Ing. habil. Boris Lohmann ist kollegialer Leiter des Instituts für Automatisierungstechnik der Universität Bremen und vertritt dort das Fachgebiet „Systemdynamik und Regelungstechnik“. Forschungsschwerpunkte liegen derzeit auf den Gebieten der Modellierung und Reduktion dynamischer Systeme, der aktiven Geräuschdämpfung im Kraftfahrzeug und der autonomen mobilen Systeme.

Adresse: Universität Bremen, Otto-Hahn-Allee NW 1, D-28359 Bremen, E-Mail: bl@iat.uni-bremen.de, www.iat.uni-bremen.de