

VOM LASERSCAN ZUM PLANUNGSTAUGLICHEN PRODUKTMODELL

M. Schleinkofer^{*}, T. Schäfer^{**}, C. van Treeck^{*}, E. Rank^{*}

** Lehrstuhl für Bauinformatik
Technische Universität München
Arcisstraße 21
80333 München
Germany*
E-mail: schleinkofer@bv.tum.de

*** Lehrstuhl für Geodäsie
Technische Universität München
Arcisstraße 21
80333 München
Germany*

Schlagwörter: Laseraufmaß, Laserscanning, Produktmodell, Bauteilerkennung, künstliche Intelligenz, virtuelle neuronale Netze

Kurzfassung: *Im Bereich der Altbausanierung und der Bestandserfassung im Bauwesen ist es häufig notwendig, bestehende Pläne hinsichtlich des Bauwerkszustandes zu aktualisieren oder, wenn diese Pläne nicht (mehr) zugänglich sind, gänzlich neue Planunterlagen des Ist-Zustandes zu erstellen. Ein komfortabler Weg, diese Bauwerksdaten zu erheben, eröffnet die Technologie der Laservermessung. Der vorliegende Artikel stellt in diesem Zusammenhang Ansätze zur Teilautomatisierung der Generierung eines dreidimensionalen Computermodells eines Bauwerkes vor. Als Ergebnis wird ein Volumenmodell bereitgestellt, in dem zunächst die geometrischen und topologischen Informationen über Flächen, Kanten und Punkte im Sinne eines B-rep Modells beschrieben sind. Die Objekte dieses Volumenmodells werden mit Verfahren aus dem Bereich der künstlichen Intelligenz analysiert und in Bauteilklassen systematisch kategorisiert. Die Kenntnis der Bauteilsemantik erlaubt es somit, aus den Daten ein Bauwerks-Produktmodell abzuleiten und dieses einzelnen Fachplanern – etwa zur Erstellung eines Energiepasses – zugänglich zu machen. Der Aufsatz zeigt den erfolgreichen Einsatz virtueller neuronaler Netze im Bereich der Bestandserfassung anhand eines komplexen Beispiels.*

1 EINLEITUNG

Umbau und Altbausanierung machen einen Großteil des Baugeschehens in Industrieländern aus. Deshalb ist es für eine langfristige Schonung von Umwelt und Ressourcen besonders wichtig, hierfür leistungsfähige, computergestützte Planungs- und Simulationswerkzeuge bereitzustellen. In einem vom Bayerischen Umweltministerium geförderten Forschungsprojekt wurden hierzu Grundlagen einer modellbasierten Planung für den Baubestand entwickelt [16]. Kern ist ein so genanntes Produktmodell, in dem neben der dreidimensionalen Geometrie eines Bauwerks alle relevanten Produktdaten (Materialien, physikalische Eigenschaften, ökologische Daten, etc.) gespeichert sind und für Simulationen des gesamten Lebenszyklus des Bauwerks genutzt werden können [8]. Der Schwerpunkt des Projekts lag auf einer weitgehend automatisierten, zweistufigen Erstellung dieses Produktmodells. Für eine erfolgreiche Planung ist eine exakte Bestandsaufnahme des aktuellen Bauzustandes unerlässlich. In einem ersten Schritt wird deshalb das geometrische Volumenmodell mit Methoden der lasergestützten Ingenieurvermessung aufgebaut. Dazu wird die erhobene Punktwolke analysiert und zunächst in ein Oberflächenmodell überführt, welches die Grundlage für die anschließende Erstellung eines Volumenmodells ist. Aus diesem wird dann in einem zweiten Schritt das Produktmodell abgeleitet.

2 BESTANDSERFASSUNG UND MODELLIERUNG

2.1 Bauaufnahme mit Methoden des Laseraufmaßes

Zunächst ist das geometrische Volumenmodell mit Methoden der lasergestützten Ingenieurvermessung – hierzu zählen das Aufmaß mit Handlasermetern, die reflektorlose Tachymetrie und die neuartige Technologie des terrestrischen Laserscannings (TLS) – aufzubauen. Dabei werden die unterschiedlichen Sensoren auf Basis von Herstellersoftwarepaketen zu einem integrierten Datenfluss kombiniert (Bild 1).

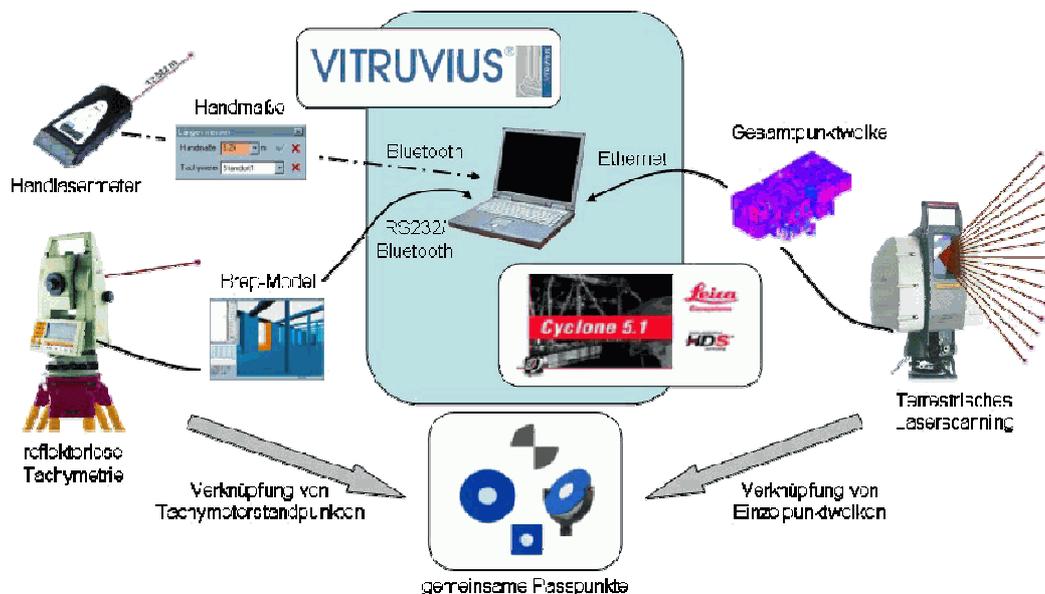


Bild 1: Verknüpfungspunkte, Datenfluss und Auswertestrategie für ein kombiniertes lasergestütztes Gebäudeaufmaßsystem

Während das Handlasermeter – als elektronische Weiterentwicklung des Zollstocks – lediglich ausgewählte Raumstrecken misst, aus denen durch Verschneidungsalgorithmen Grundrisspläne erstellt werden können, bietet die reflektorlose Tachymetrie durch Messung von Horizontal- und Vertikalwinkeln sowie der Raumdistanz zu nicht signalisierten Zielen die Möglichkeit, beliebige bzw. charakteristische Objektpunkte anzumessen. Insbesondere durch das Laserscanning ist heute jedoch auch eine effektive und vollständige dreidimensionale Abbildung komplexester Umgebungen möglich [26]. Das 3D-Laserscanning adaptiert dabei das Prinzip der Polaraufnahme der elektronischen Tachymetrie, ist aber durch eine automatisierte Verstellung von Umlenkspiegeln und somit des messenden Laserstrahls in der Lage, eine große Anzahl von Einzelpunkten in kürzester Zeit und mit einer nahezu beliebigen Auflösung zu registrieren. Ergebnis dieser Erfassung ist eine dreidimensionale Punktwolke (Bild 2), die aus mehreren Millionen Einzelpunkten bestehen kann.

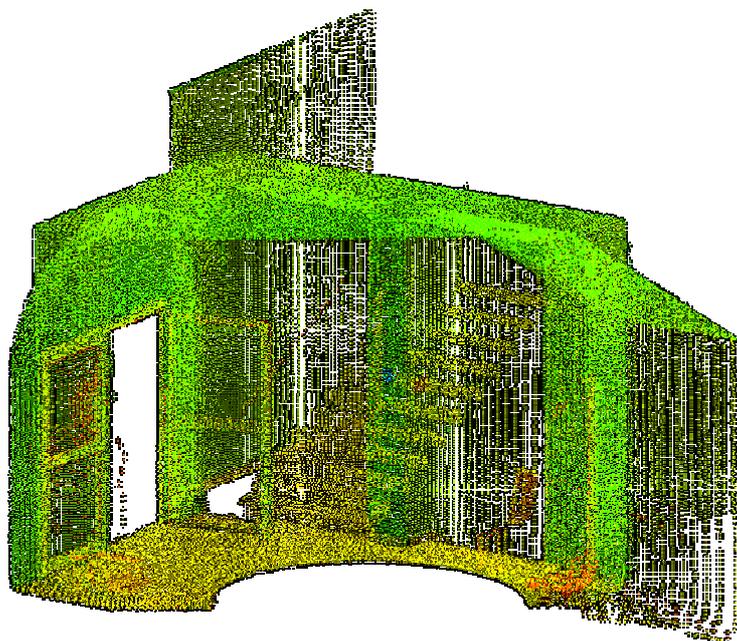


Bild 2: Ausschnitt einer Laserscan-Punktwolke eines Bürogebäudes. Die Farbgebung der Punkte spiegelt die rückgestreute Laserintensität wider

2.2 Modellierung

Eine mit Hilfe der Laservermessung erhaltene Punktwolke enthält umfassende Information über die Geometrie in Form einzelner kartesischer Punkte, welche in hoher Dichte vorliegen. Eine Aussage über den Zusammenhang einzelner Flächen und deren Topologie ist in diesem Stadium aber nicht zu treffen. Für eine Weiternutzung sind Flächen und Körper zu modellieren, was einen erheblichen Aufwand darstellt [10]. Verfügbare Modellierungswerkzeuge konzentrieren sich deshalb meist auf die semiautomatische Ableitung von Objekten wie beispielsweise Ebenen, Zylinder (Abb. 3) und Quader. In diesem Fall ist jeder zu modellierende Punktwolkenausschnitt vom Anwender explizit zu segmentieren und in ein Flächen- bzw. Volumenobjekt umzuwandeln. Ein weiterer Ansatz beruht auf dem Prinzip des Bereichswachstumsverfahrens (region growing) [25], bei dem der Anwender einen geeigneten Saatpunkt in der Punktwolke des zu modellierenden Bereiches auswählt und anschließend über Nachbarschaftsbeziehungen ein zuvor festgelegtes Objekt segmentiert wird.

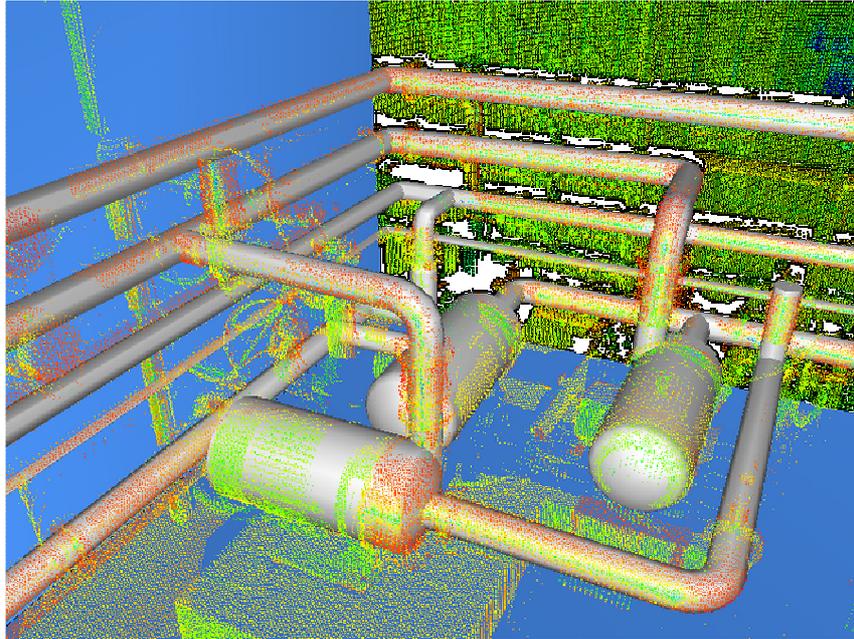


Bild 3: Punktwolke und daraus abgeleitete Rohrleitungen in einer industriellen Anlage

Zur Weiterentwicklung dieser Modellierungsansätze wurde deshalb von den Autoren ein Verfahren zur automatischen Extraktion von vertikal ausgerichteten Ebenen innerhalb einer Punktwolke entwickelt und prototypisch am Beispiel eines unmöblierten Stockwerks mit Büroräumen erfolgreich eingesetzt [12]. Grundlage des Verfahrens bilden die aufnahmebedingten Unterschiede in den Punktdichten von Punktwolken, die jeweils in die Horizontalebene projiziert werden. So liegt die Anzahl der Punkte pro Grundfläche auf einer vertikalen Ebene, die üblicherweise mit einigen Zentimetern Auflösung gescannt wird, deutlich höher als beispielsweise auf einem Fußboden. Mit Ansätzen aus der Digitalen Bildverarbeitung [6] lassen sich aus einem binären Punktdichtebild ebenfalls Kantenpolygone ableiten (Bild 4), die als potenzieller Suchraum (Bild 5) für Ebenen in der dreidimensionalen Punktwolke dienen können.

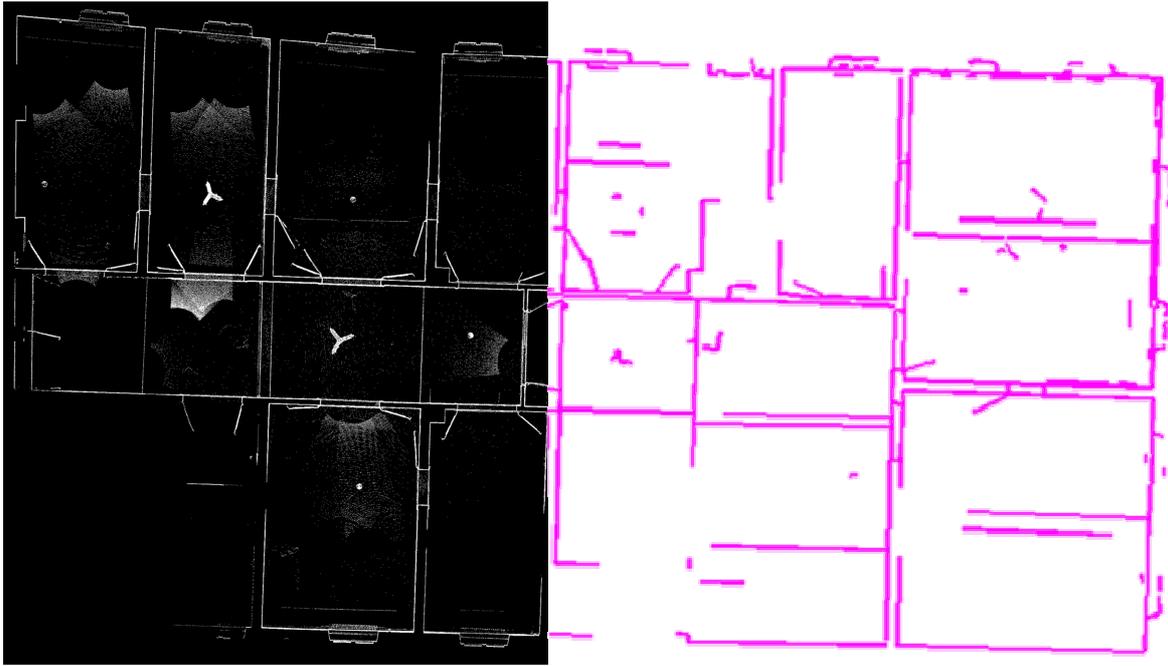


Bild 4: Methoden der Digitalen Bildverarbeitung: Aus Punktdichtebild extrahierte Kantenpolygone

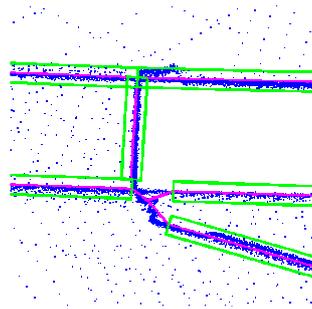


Bild 5: Abgeleitete Suchräume für Vertikalebenen

Mit Hilfe dieser alternativen Vorgehensweise können somit einzelne, im Modell enthaltene vertikale Ebenen extrahiert werden. Die Berechnung erfolgt dabei nach der Methode der kleinsten Fehlerquadrate [18], indem eine jeweils durch eine Menge von Punkten gelegte Ebene der Forderung genügen muss, das Abstandsquadrat zu diesen Punkten zu minimieren. In Bild 6 wird dieses Vorgehen anhand der Punktwolke eines Gebäudestockwerkes demonstriert, indem aus als zusammengehörig identifizierten Punkten zunächst einzelne Ebenen erzeugt werden. Die dargestellten Flächen ergeben sich anschließend aus der lokalen Begrenzung der zugeordneten Punktwolken.

Die nächste Entwicklungsstufe wird die Herleitung eines vollständigen Oberflächen- und zukünftig Volumenmodells sein. Eine Automatisierung dieses Arbeitsschrittes erfordert zunächst ein Prioritätenkonzept zur sinnvollen Verschneidung von Ebenen. Grundlage für ein solches Konzept könnten dabei die Ebenenparameter darstellen – somit ließen sich beispielsweise parallele Ebenen mit einem Abstand von 15 cm zu einer Wand zugehörig klassifizieren.

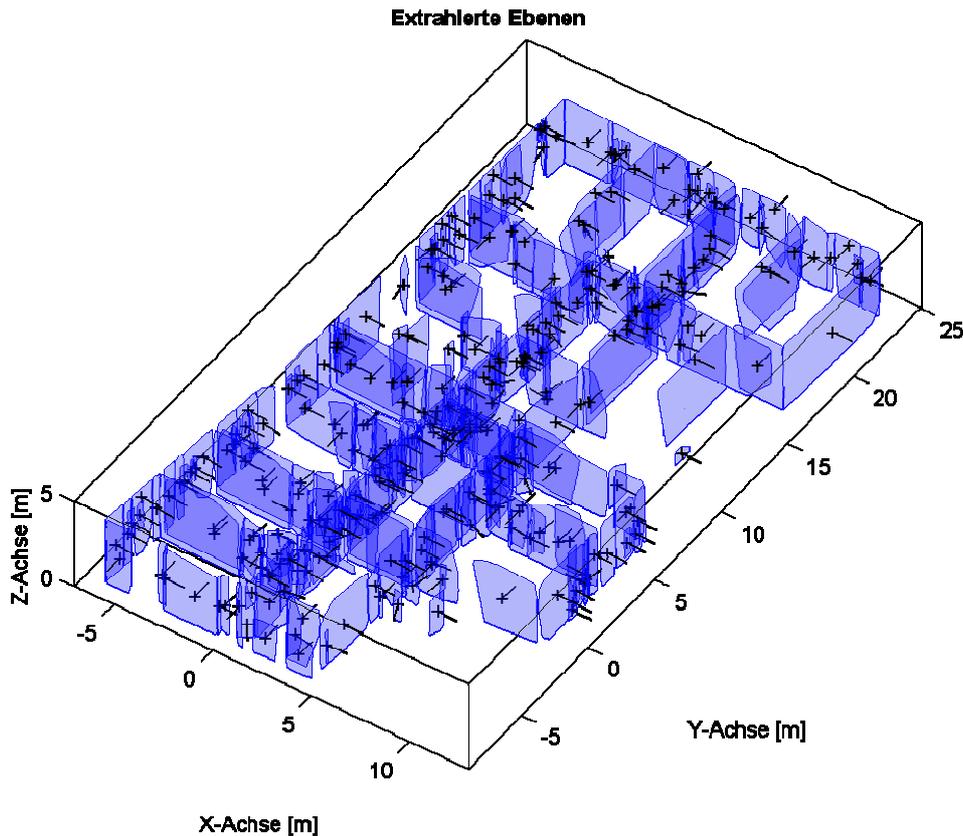


Bild 6: Ergebnis der automatischen Modellierung von 370 Ebenen aus einer Gesamtpunktwolke mit $6 \cdot 10^6$ Einzelpunkten

3 NEURONALE NETZE ZUR BAUTEILKLASSIFIZIERUNG

Die Überführung des Volumenmodells in ein Produktmodell mit AEC-Objekten (Architecture, Engineering and Construction) erfolgt automatisch. Dafür ist es erforderlich, dass der Computer die Objekte des aus Abschnitt 2 erhaltenen Volumenmodells als Bauteile identifiziert. Während in anderen Bereichen die Mustererkennung schon sehr weit fortgeschritten ist (z. B. Bildanalyse, Biometrie) [27], existieren im Bereich Bauwesen bislang wenige Ansätze zur Identifizierung von Bauteilen. Um Bauteile, die als topologisch eindeutige Volumenobjekte vorliegen, zu erkennen, ist es nahe liegend, deren charakteristische Merkmale zur Entscheidungsfindung heranzuziehen. Insbesondere geometrische Eigenschaften wie Hauptachsen, Flächenmomente, Oberflächenzahlen und -ausdehnung, Raumorientierung, etc. erscheinen Erfolg versprechend. Diese Attribute müssen zur Auswertung in eine algorithmisch verständliche Form gebracht werden.

Methoden der künstlichen Intelligenz bieten die Möglichkeit, verschiedene Kriterien auszuwerten und – im Gegensatz zu „harten“ Regeln eines Entscheidungsbaumes – die Ergebnisse als Gewichtungen für eine Entscheidung darzubieten. Eventuelle Fehlentscheidungen, die durch „knappes Nichterfüllen“ von einzelnen Anforderungen entstehen würden, werden ausgeglichen, so dass die Gesamtheit der Bewertungen in der Regel dennoch eine korrekte Entscheidung zulässt. Für die Lösung des gegebenen Problems werden virtuelle neuronale Netze vorgestellt. Die Prägung des Netzes in der Trainingsphase bestimmt dabei die Güte des Verfahrens und ist deshalb als Hauptaufgabe zu verstehen.

3.1 Virtuelle neuronale Netze

Virtuelle, auch künstliche neuronale Netze (vnn) bestehen aus Zellen (Neuronen), die miteinander in Kontakt stehen [14]. Jede dieser Zellen wertet die an sie übergebenen Eingangssignale aus. Die daraus ermittelte Aktivierung der Zelle wird als Ausgabesignal an andere Zellen geschickt (Bild 7). Die einfachste Form vernetzter Neuronen, welche nur die Zustandsgrößen 0 und 1 annehmen können, wird Perzeptron genannt [15]. Die Eingangswerte des Neurons bestimmen sich aus der Aktivierung x_i des Vorgängerneurons i , welche mit einem Gewichtungsfaktor $w_i \in \mathfrak{R}$ multipliziert und somit verstärkt oder geschwächt wird. Überschreitet die Summe dieser Produkte einen Schwellwert, so schaltet die Aktivierung der Zelle auf 1, andernfalls verbleibt sie auf 0.

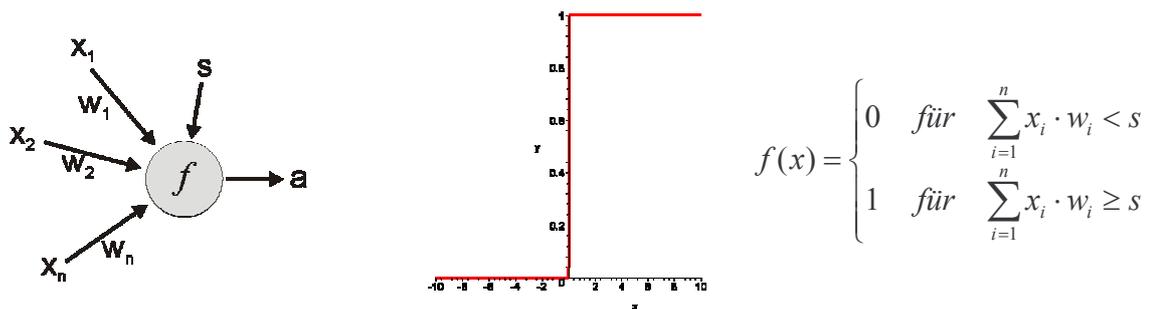


Bild 7: Perzeptron mit Eingangssignalen $x_i \cdot w_i$ und Schwellwert s , Beschreibung der Aktivierungsfunktion mit Ausgabesignal a (Werte 0 oder 1)

Während in der beschriebenen Form die Aktivierung des Neurons nur zwei Zustände erreichen kann, erlaubt eine differenzierbare Aktivierungsfunktion das Erlernen von Mustern mit der Back-Propagation-Regel [13]. Das Übertragen des Schwellwertes auf die Eingabeseite vereinfacht die weitere Betrachtung. Dazu wird für jede Zelle ein zusätzliches Eingabeneuron mit der konstanten Aktivierung 1 eingeführt, wodurch der ursprüngliche Schwellwert durch die Gewichtung $-w_{n+1}$ repräsentiert wird (Bild 8).

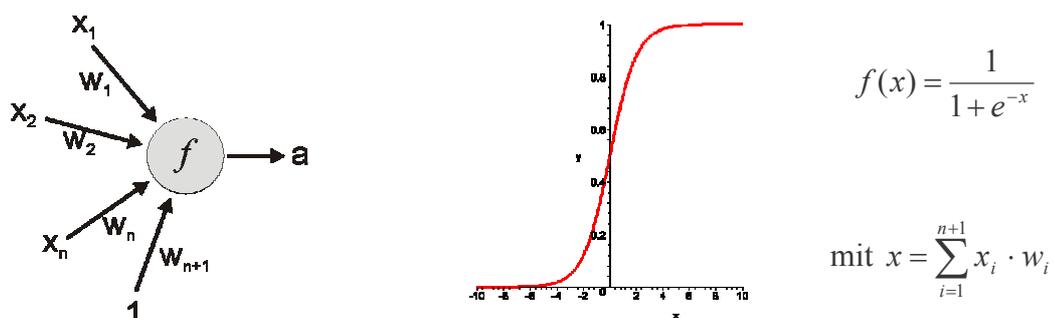


Bild 8: Neuron mit Eingangssignalen $x_i \cdot w_i$ und Schwellwert w_{n+1} , differenzierbarer Aktivierungsfunktion mit Ausgabesignal a im Wertebereich $]0; 1[$

Einem neuronalen Netz sind Informationsdaten zu übergeben, welche dieses intern auswertet um schließlich das Berechnungsergebnis zu präsentieren. Für die Kommunikation des Benutzers mit dem Netz stehen Eingabeneuronen (Eingabeschicht) und Ausgabeneuronen (Ausgabeschicht) zur Verfügung. Alle dazwischen liegenden Neuronen gehören verborgenen Schichten an, welche dem Zugriff durch den Benutzer entzogen sind. Es gibt eine große Vielzahl an unterschiedlichen neuronalen Netzen, die sich in ihrem Verbindungsaufbau, der Propagierungsregel und dem Lernalgorithmus unterscheiden, siehe z. B. in [27]. Ein einfaches

Netz ist das ebenenweise verbundene vorwärtsgerichtete Netz (Bild 9), das auch zur Lösung des gegebenen Problems verwendet wird.

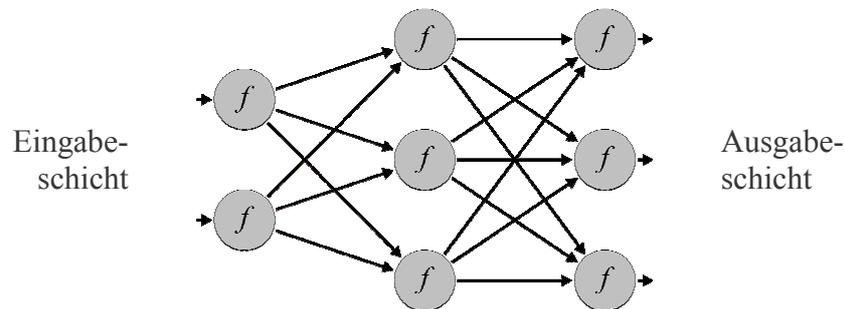


Bild 9: Virtuelles neuronales Netz, hierarchisch strukturiert mit ausschließlich vorwärtsgerichteter Signalübermittlung

Das „Wissen“ eines trainierten Netzes steckt in seinen Kantengewichten, die die Weitergabe der Signale zwischen den Neuronen jeweils verstärken oder schwächen. Während des Lernvorgangs werden diese Gewichte so modifiziert, dass die Ausgabe für einen Trainingsdatensatz optimiert wird. Ziel des Trainings ist, dass das konditionierte Netz später ähnliche, aber unbekannte Datenmuster erkennen kann. Bei zu starker Prägung auf die Trainingsdaten ist dies nicht mehr möglich, man spricht dann vom „auswendig Lernen“ des Netzes (siehe 3.8).

3.2 Vorgehensweise

Die Ausgangsbasis für die Bauteilklassifizierung sind die 3D-Volumenkörper, die das Ergebnis der Modellierung der vermessungstechnisch aufgenommenen Objektgeometrie sind (siehe 2.2). Diese können prinzipiell beliebig geformt sein, so dass es notwendig ist, sich Gedanken zu machen, wie für diese Volumenkörper aussagekräftige Parameter erhoben werden können. Sind die Eigenschaften ausgewählt, werden sie in eine dem künstlichen neuronalen Netz zugängliche Form gebracht. Das Netz, das auf der Ausgabeseite eine der Menge an zu erkennenden Bauteilklassen entsprechende Anzahl an Ausgabeneuronen besitzt, liefert für jede Klasse einen Ausgabewert, der jeweils als Netzentscheidung zu interpretieren ist (siehe 3.5).

3.3 Auswahl der Geometrieattribute

Künstliche neuronale Netze werden zur Mustererkennung für 2D-Bildmuster bereits erfolgreich eingesetzt [14][20]. Dabei werden die Helligkeitswerte von Farbkomponenten (alternativ auch nur für Graustufen) an ein neuronales Netz übergeben. In Analogie daran wäre eine Möglichkeit für das hier gestellte Problem gewesen, die im kontinuierlichen Raum befindlichen Objekte in einem Raster zu diskretisieren („voxeln“) [21] und die resultierende Voxelmatrix an das Netz zu übergeben. Es ist jedoch zu bedenken, dass die Größe der Eingabeschicht von der Anzahl der Voxel abhängt. Diese Anzahl – und somit die Netzgröße – wäre bei entsprechend feiner Rasterauflösung sehr groß und würde ein immenses Datenaufkommen implizieren. Das hier gezeigte Verfahren geht einen anderen Weg und verwendet die Geometrieattribute der Objekte direkt.

Um Bauteile identifizieren zu können ist es notwendig, Entscheidungsparameter festzulegen, die für alle Objekte gleichermaßen aussagekräftig sind. Unsere Herangehensweise dabei ist es, Attribute zu verwenden, die auch ein Mensch als Zuordnungskriterien verwenden würde. Dabei wird versucht, die Objekte weitgehend

unabhängig von ihrer Lage im Gesamtbild auszuwerten. Nachbarschaftsbeziehungen zu anderen Körpern werden deshalb nur bedingt berücksichtigt.

Es werden folgende geometrische Eigenschaften der Volumenkörper als Entscheidungskriterium herangezogen:

- Verwendet wird die Ausdehnung des Objektes. Die Höhe wird direkt abgegriffen, die beiden horizontalen Dimensionen werden mittels einer kleinsten umschließenden Box bestimmt [11]. Dazu wird der Körper auf die Horizontalebene projiziert.
- Des Weiteren werden die Körperoberfläche und das Volumen ermittelt.
- Mit Bool'schen Operationen wird die Lage innerhalb anderer Körper festgestellt. Hier gilt anzumerken, dass nur Bauteile, also massebehaftete Elemente als Volumenkörper repräsentiert sein dürfen, nicht aber topologische Bereiche wie der Raum selbst. Herausschneidende Öffnungen haben in diesem Zusammenhang aber eine Sonderstellung: Deren übergeordnete Volumenelemente (z. B. die späteren Wände) werden ohne Öffnungen dargestellt. Da die masselosen Öffnungen selbst Bauteile repräsentieren (Fenster, Türen), sind diese bereits im Volumenmodell als eigene Volumenkörper darzustellen (Bild 10).
- Für Körper, die innerhalb anderer liegen – in der Praxis sind dies z. B. Fenster in Wänden – wird der vertikale Abstand der beiden Unterkanten – im Beispiel also die Simshöhe – ermittelt.

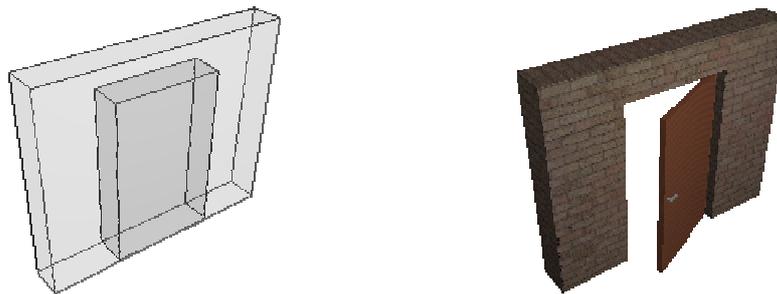


Bild 10: Eine Wand mit einer darin befindlichen Tür wird im vorliegenden Volumenmodell als ineinander liegende Volumenkörper repräsentiert

3.4 Aufbereitung der Geometrieattribute

Bei der Suche nach den passenden Eingangsgrößen für das Netz wurde festgestellt, dass die Übergabe aller verfügbaren Parameter nicht zwangsläufig zu besseren Ergebnissen führt. Dies lässt sich dadurch begründen, dass zusätzliche Eingabeparameter zu zusätzlichen Neuronen – zunächst in der Eingabeschicht – führen, welche wiederum die Anzahl an Freiheitsgraden im Netz erhöhen. Diesem müsste mit einer Erweiterung der Menge an unterschiedlichen Trainingsdaten begegnet werden, was nicht selbstverständlich möglich ist. Ferner stellte sich heraus, dass sich das Netz durch Verhältniswerte von Größen zueinander besser trainieren lässt als durch die Angabe absolut gegebener Werte. Somit ergab sich als sinnvolle Auswahl an Netzeingangswerten:

- x / y : Verhältnis der größeren zur kleineren horizontalen Ausdehnung
- x / z : Verhältnis der größeren horizontalen zur vertikalen Ausdehnung

- “liegt innerhalb von”: Anzahl an Objekten, innerhalb derer der aktuelle Festkörper liegt (in einem konsistenten Modell 0 oder 1)
- z-Abstand: Abstand zwischen der minimalen z-Kote des aktuellen Objektes und dessen umschließenden (sofern vorhanden)

Da Null-Signale keinen Zugewinn zum Lernvorgang leisten, wird das Attribut „liegt innerhalb von“ statt der realen vorkommenden Werte 0 und 1 – wie in [19] vorgeschlagen – auf -1 und +1 gesetzt.

Die Übergabe an das virtuelle neuronale Netz erfolgt über einen Vektor von Fließkommazahlen. Dementsprechend müssen alle Attribute in entsprechender Form an diesen Vektor übergeben werden.

3.5 Auswertung der Entscheidung des neuronalen Netzes

Die Anzahl an Neuronen in der Ausgangsschicht bestimmt sich durch die Anzahl an zu unterscheidenden Bauteilklassen. Die Bauteilkategorie, dessen zugeordnetes Neuron (siehe dazu auch Bild 12) den größten Aktivierungszustand aufweist, wird als Netzentscheidung gewertet. Im gegebenen Fall sollen die in Bild 11 gezeigten sieben Bauteile unterschieden werden.

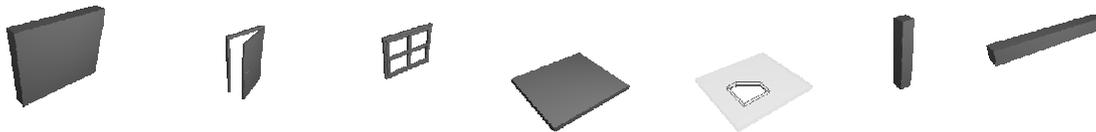


Bild 11: unterschiedene Bauteilklassen (Wände, Türen und Fenster, Deckenplatten und – durchbrüche, Stützen und Balken)

3.6 Netztopologie

Durch die vorgestellten Randbedingungen Eingabeparameter (siehe 3.4) und unterscheidbare Bauteilklassen (siehe 3.5) ist die Anzahl an Eingabeneuronen auf vier und die der Ausgabeneuronen auf sieben festgelegt. Nun ist durch eine geeignete Wahl an verdeckten Neuronen ein virtuelles neuronales Netz zu erzeugen, das eine Bauteilklassifizierung möglichst präzise ermöglicht.

Zu große Netze sind schwer zu trainieren: Die große Zahl an Netzkantengewichten und –schwellewerten muss während des Trainings mit einer umfassenden Menge an unterschiedlichen Trainingsdaten festgelegt werden. Ferner tendieren unnötig große Netze dazu, sich auf die Trainingsmuster zu spezialisieren und ihre Flexibilität gegenüber unbekanntem Mustern zu verlieren [1]. Wird das Netz auf der anderen Seite zu klein gewählt, ist es nicht ausreichend trainierbar. Da es keine deterministischen Regeln für die passende Topologie gibt, ist es konzeptionell schwierig, ein geeignetes Netz festzulegen. Wie durchgeführte Untersuchungen ergaben, besitzt für die gegebene Problemstellung das günstigste Netz eine einzelne verborgene Schicht, welche ebenso viele Neuronen wie in der größeren der beiden außen liegenden beinhaltet. In Bild 12 ist das verwendete Netz mit seinen Ein- und Ausgabeparametern dargestellt.

Objektparameter

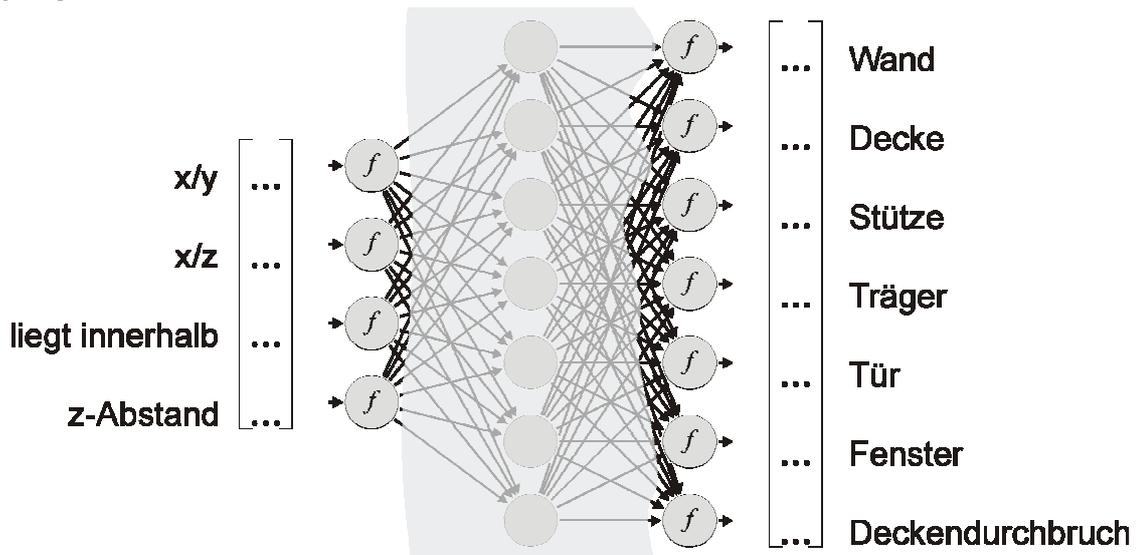


Bild 12: Eingabe- und Ausgabestruktur des verwendeten virtuellen neuronalen Netzes zur Bauteilerkennung

3.7 Vorbereitung der Trainingsdaten

Um ein virtuelles neuronales Netz erfolgreich zu trainieren, müssen bereits bekannte Datensätze erlernt werden. In diesem Zusammenhang spricht man von einzelnen Trainingsmustern, welche zusammen die Trainingsbasis bilden. Die Netzeingabewerte (auch Merkmalsvektor [7]) spannen den Merkmalsraum auf. Die zum Training verwendeten Trainingsmuster sollen sich möglichst gleichmäßig im Merkmalsraum, insbesondere auch an den Klassifizierungsgrenzen verteilen. Identische Datensätze sind dabei zu vermeiden. Um das Netz für alle Ergebnisfälle gleichermaßen zu trainieren, muss für alle gewünschten Klassen eine ausreichende Menge an Mustern zur Verfügung stehen.

Es ist nahe liegend, bestehende Produktmodelle als Trainingsbasis zu verwenden. Einzelne Bauobjekte weisen hierbei jedoch oft nicht die notwendige Vielfalt in den einzelnen Bauteilen auf (Bild 13). Um dennoch eine ausreichend große Anzahl an Bauteilen zu erhalten, können beliebig viele Produktmodelle betrachtet und der Trainingsbasis hinzugefügt werden.

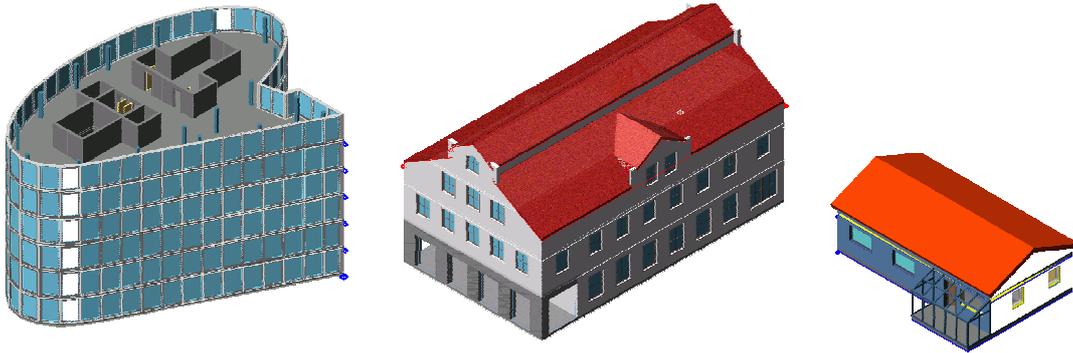


Bild 13: Einzelne Bauobjekte weisen eine nicht ausreichende Varianz in den einzelnen Bauteilen auf

Bild 13-1: Moderne Hochhäuser oder Baukomplexe wiederholen sich in den Stockwerken, wiederkehrende Bauelemente innerhalb der Geschosse wie Stützen und Fenster

Bild 13-2: Altbauten besitzen eine größere Fülle an unterschiedlichen Bauteilen, jedoch werden auch hier häufig Elemente mit gleichen Maßen verwendet (Fenster, übereinander liegende Wände)

Bild 13-3: Einfamilienhäuser sind häufig sehr individuell gestaltet, was sich auch in den Einzelbauteilen zeigt, jedoch ist die Anzahl an Bauteilen gering

3.8 Durchführung des Trainings

Das Netz entscheidet sich für eine Bauteilklasse anhand des Ausgabeneurons, das die größte Aktivierung besitzt (siehe 3.5). Bedingt durch die der Propagierungsvorschrift zugrunde liegende Aktivierungsfunktion (Bild 7) liegt die Aktivierung eines Neurons im Bereich $]0; 1[$. Ziel des Trainings ist es, dass für jeden Merkmalsvektor der Bauteile stets das korrelierende Ausgabeneuron die dominierende Aktivierung besitzt, im Optimalfall 1 für das gewünschte Neuron und 0 für alle anderen.

Um zu verhindern, dass die Trainingsbasis die veränderlichen Größen des Netzes unbemerkt „überlernt“, wird ein Anteil an Trainingsmustern aus der Trainingsbasis herausgenommen. Dieser wird nicht zum Trainieren, sondern ausschließlich für die Bewertung des Lernzustandes des Netzes verwendet. Steigt nach einer Anzahl an Trainingszyklen der Prozentsatz an Fehlklassifizierungen in dieser Referenzgruppe wieder an, so ist dies ein Indiz dafür, dass das Netz seine Fähigkeit zur Verallgemeinerung verloren hat und auf die vorgelegten Trainingsmuster geprägt ist (Bild 14). Das Training ist deshalb zuvor abzubrechen.

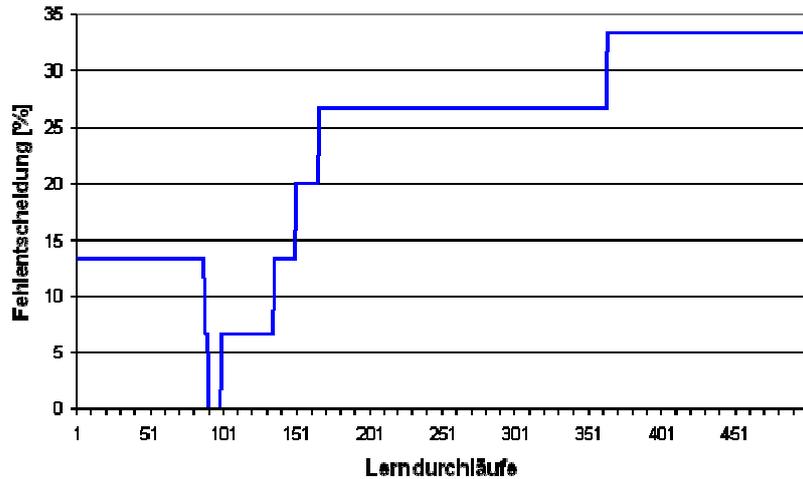


Bild 14: Entwicklung der Fehlentscheidungen über die Lerndurchläufe (Beispiel: Lernen der Farberkennung anhand von RGB-Werten mit zu kleiner Trainingsbasis und zu vielen Trainingszyklen)

3.9 Umsetzung in einer Beispielanwendung

Um das vorgestellte Verfahren praxisnah zu testen, wurde das Klassifizierungsmodul in die kommerzielle CAD-Software Architectural Desktop (ADT) integriert. Dessen Hersteller Autodesk bietet zur Steuerung und Erweiterung der Funktionalität von AutoCAD sowie dessen Architektur-Erweiterung ADT Programmierschnittstellen an (Bild 15). Damit ist es möglich, innerhalb der CAD-Umgebung bestehende Produktmodelle zu zerlegen und als Trainingsobjekte zu verwenden.

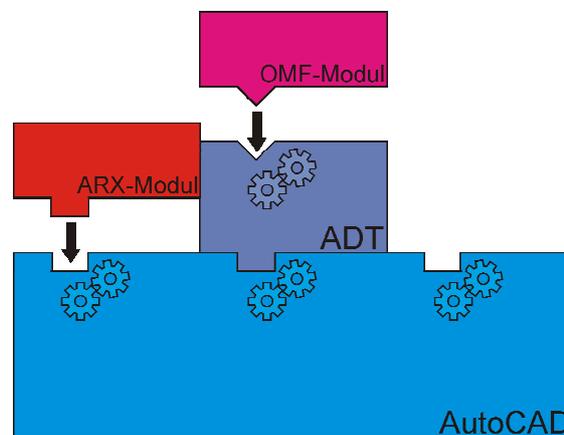


Bild 15: Programmierschnittstellen für Autodesk AutoCAD (AutoCAD runtime extension programming environment, kurz ObjectARX) sowie zur Erweiterung Architectural Desktop (ADT) (Autodesk AEC Object Modeling Framework, kurz OMF)

Mit Hilfe der Daten aus dem Laseraufmaßes werden, wie in Abschnitt 2.2 beschrieben, Standardkörper erstellt. Im CAD-System kann nun die Geometrie dieser Körper ausgelesen werden. Anhand der bereits vorgestellten Attribute (siehe 3.3) klassifiziert das trainierte virtuelle neuronale Netz die Objekte, welche für das Produktmodell in die entsprechenden Bauteilklassen umgewandelt werden. Dem Anwender stehen somit „intelligente“ Objekte zur Verfügung, welche z. B. physikalische Kennwerte besitzen können. Ein möglicher

Anwendungsfall ist die Studie verschiedener Varianten in der Energieoptimierung [5][21], welche den Energieaufwand des Gebäudes aus den Eigenschaften der einzelnen Bauteile ableitet.

Das beschriebene Verfahren wurde im Rahmen eines Forschungsprojektes an einem konkreten Objekt angewendet. Zunächst wurden bestehende Produktmodelle von drei verschiedenen Objekten – eines Bürohochhauses, eines Geschäftsgebäudes und eines Einfamilienhauses (Bild 13) – verwendet, um die Datenbasis für die Trainingsphase eines neuronalen Netzes zu erstellen. Obwohl die Anzahl an unterschiedlichen Objekten als zu gering zu bezeichnen ist, war es dennoch möglich, mit den 87 verfügbaren Bauteilen das Netz für den anschließenden Anwendungsfall zu trainieren. Aus den Laserscan-Daten zweier Stockwerke aus einem der Gebäude des „Alten Hofes“ in München (siehe [24]), welche zur Zeit des Aufmaßes als Bürogebäude genutzt wurden, wurde ein Volumenmodell mit etwa 100 Objekten erstellt. Dieses konnte ohne Fehler automatisch in ein Produktmodell mit den zugehörigen Bauteilen überführt werden (Bild 16).

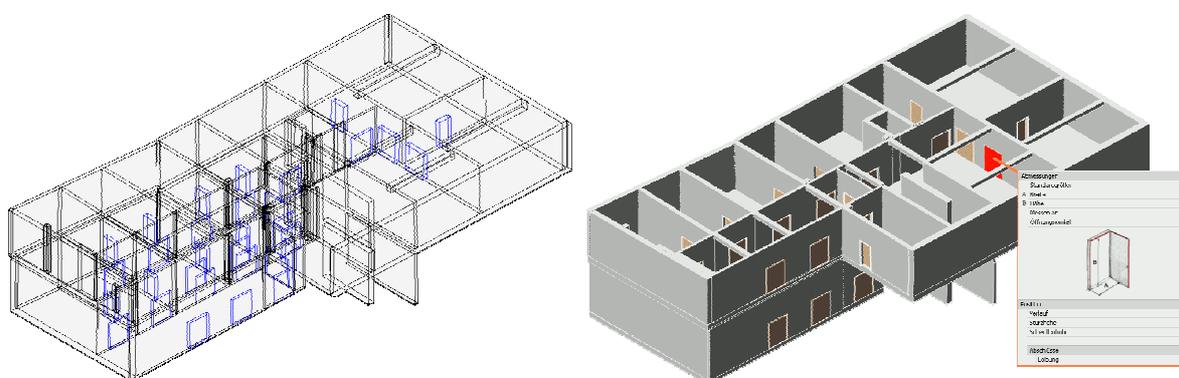


Bild 16: Drahtgittermodell des aus Laserscan-Daten gewonnenen Volumenmodells mit einheitlichen Volumenobjekttypen sowie daraus abgeleitetes Produktmodell mit unterschiedlichen Bauteilklassen

4 AUSBLICK

Ein Resultat der bisherigen Forschungsarbeiten ist, dass aus den aufgenommenen Laserscan-Punkten oft nur mit großem Aufwand Volumenobjekte zu rekonstruieren sind. Deshalb gehen weiterführende Überlegungen von einer Unterstützung der Volumenmodellgenerierung durch zusätzliche Informationsquellen aus. Als Möglichkeiten bieten sich bestehende (zweidimensionale) Planunterlagen oder vor Ort erstellte Skizzen an. Sind die Papierskizzen in digitaler Form aufbereitet worden, werden sie mit den gescannten Laserkoordinaten abgeglichen und können so die Erstellung der dreidimensionalen Objekte aus den vermessenen Raumpunkten erheblich erleichtern. Positive Erfahrungen konnten dabei in einer verwandten Anwendung aus der Photogrammetrie gewonnen werden: Aus boden- und luftgestützten Laseraufnahmen werden heute unter Zuhilfenahme von Bebauungsplänen erfolgreich virtuelle Stadtmodelle erzeugt [4].

Neu erstellte Skizzen sollten bereits digital angefertigt werden und können durch notwendige Daten wie Bauteilklassenzuordnung ergänzt werden. In diesem Falle ist lediglich die Anpassung an die exakte, aufgemessene Geometrie notwendig, um ein planungstaugliches Produktmodell zu erhalten. Prototypen zu diesem Vorgehen sind in [20] beschrieben, bei dem

dreidimensionale Objekte am Computer entworfen und durch Laseraufmaß in ihrer Geometrie festgelegt werden.

Findet die Unterstützung hingegen durch bestehende Papierpläne oder Handskizzen statt, sind diese zunächst für sich zu analysieren [2]. Erkannte Linien können dabei helfen, den Suchbereich wie schon in 2.2 gezeigt entscheidend einzuschränken und Oberflächen aus der Punktwolke zu extrahieren (Bild 17). Die erkannten Topologien beschreiben ferner zusammengehörige Oberflächen, die zu geschlossenen Volumenobjekten modelliert werden. Diese sind mit dem bereits erwähnten Verfahren der künstlichen neuronalen Netze in Bauteilklassen und somit in das angestrebte Produktmodell überführbar.

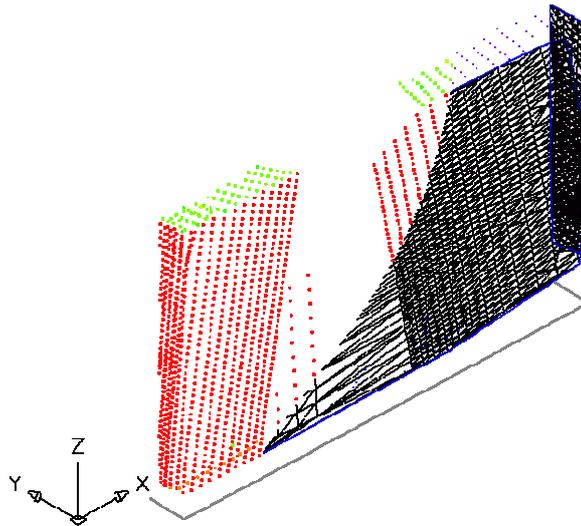


Bild 17: In die Punktwolke (grün, rot) eingefügte Skizzenlinien (grau) unterstützen die Erstellung vernetzter (schwarz) Flächen (blau)

Zur Erkennung und Extraktion von Zusammenhangskomponenten sei in diesem Zusammenhang auch auf die weiteren Vorarbeiten der Forschergruppe [22] verwiesen. Gegenwärtige Arbeiten der Gruppe konzentrieren sich auch auf die formale Definition einer räumlichen Anfragesprache (spatial query language) für Gebäude-Informationsmodelle [3].

5 ZUSAMMENFASSUNG

Die Bestandserfassung von Gebäudegeometrien mittels der drei Laseraufmaßsysteme Handlasermeter, reflektorloser Tachymetrie und terrestrischem Laserscanning (TLS) stellt eine praxisnahe und wirtschaftliche Kombination dar. Für die Weiterverarbeitung von dreidimensionalen kartesischen Punktwolken, welche bei der Geometrierfassung mit TLS-Systemen entstehen, wurde ein alternatives Modellierungsverfahren vorgestellt. Dieses ermöglicht eine teilautomatisierte Extraktion von Wandoberflächen, indem eine Punktwolke in eine Horizontalebene projiziert und in dem entstandenen Bild durch Kantenerkennungsalgorithmen der Suchraum für potenzielle Wandflächen definiert wird. Eine solche Herangehensweise reduziert den Modellierungsaufwand gegenüber herkömmlichen Segmentierungsansätzen erheblich. Das auf Methoden der Digitalen Bildverarbeitung basierende Verfahren konnte schließlich an einem Demonstrationsbauwerk erprobt und erfolgreich umgesetzt werden.

Der Einsatz virtueller neuronaler Netze in der Bauteilerkennung ist ein Weg, flexibel auf verschiedene Muster zu reagieren. Neue, dem System unbekannt Formen von Bauteilen

scheiden mit diesem Ansatz nicht von vornherein aus. Hauptaufgabe ist die Auswahl geeigneter Eingangsdaten sowie die sinnvolle Konzeption des Netzes. Attribute, die auch für den Menschen aussagekräftig sind, wurden hierfür in eine für das neuronale Netz verwertbare Form aufbereitet.

Die prototypische Anwendung der entwickelten Algorithmen konnte anhand eines komplexen Referenzgebäudes erfolgreich demonstriert werden. Standardformen von Bauteilen wurden mit sehr großer Zuverlässigkeit erkannt und den Bauteilklassen zugeordnet. Dabei können sowohl die durch Laseraufmaßverfahren erstellten Modelle als auch bereits existierende digitale 3D-Pläne in ein Produktmodell überführt werden.

Durch die vorgestellten Werkzeuge wird es möglich, auch im Bereich der Altbausanierung von den Vorteilen einer Produktmodell basierten Planung nachhaltig zu profitieren und diese bereits in frühen Planungsstadien einer Sanierungsmassnahme gewinnbringend einzusetzen. Als beispielhafte Anwendung sei die Simulation des Ressourcenverbrauches eines Gebäudes [5] genannt, die auf einem Produktmodell aufsetzt und die energetische und ökologische Bewertung und Optimierung etwa einer Instandsetzungsmaßnahme eines Gebäudes gestattet [5][8][9][12].

LITERATUR

- [1] Bastian, A.: Bauteilklassifizierung mit Hilfe virtueller neuronaler Netze. Diplomarbeit am Lehrstuhl für Bauinformatik, Technische Universität München, München 2005
- [2] Berkahn, V.; Kinkeldey, C.; Schleinkofer, M.; Rank, E.: Re-engineering based on Construction Drawings - From Ground Floor Plan to Product Model. In: Xth International Conference on Computing in Civil and Building Engineering (ICCCBE-X), S. 162-163, Weimar, Germany: Bauhaus Universität Weimar - Universitätsverlag 2004
- [3] Borrmann, A.; van Treeck, C.; Rank, E.: Towards a 3D spatial query language for building information models, International Conference on Computing and Decision Making in Civil and Building Engineering, Montreal, 14. – 16.06.2006
- [4] Brenner, C.: Dreidimensionale Gebäuderekonstruktion aus digitalen Oberflächenmodellen und Grundrissen. Dissertation am Institut für Photogrammetrie der Universität Stuttgart, Stuttgart 2000
- [5] Egger, M.; Neuberg, F.: Ökologische Bewertungsmethoden auf der Grundlage eines Bauwerkmodells. In: Forum Bauinformatik, S. 250-259, Shaker Verlag, Aachen 2003
- [6] Jähne, B.: Digitale Bildverarbeitung, Springer, Berlin 2002
- [7] Niemann, H.: Klassifikation von Mustern, Springer-Verlag, Berlin 1983
- [8] Neuberg, F.: Ein Softwarekonzept zur Internet-basierten Simulation des Ressourcenbedarfs von Bauwerken. Dissertation, Technische Universität München, Januar 2004, Online: <http://tumb1.biblio.tu-muenchen.de/publ/diss/bv/2004/neuberg.html>
- [9] Neuberg, F.; Ekkerlein, C.; Rank, E.; Faulstich, M.: Integrated Life Cycle Simulation and Assessment of Buildings. In: Xth International Conference on Computing in Civil and Building Engineering (ICCCBE-X), Weimar, Germany 2004

- [10] Niemeier, W.: Modellierung – Ableitung geometrischer Informationen aus Punktwolken. In: Barth, W., Foppe K., Schäfer, T. (Hrsg.): Terrestrisches Laserscanning (TLS) - Ein geodätisches Messverfahren mit Zukunft, Beiträge zum 65. DVW-Seminar am 21. und 22. November 2005 in Fulda, Schriftenreihe des DVW, Band 48, Wißner-Verlag, Augsburg 2005
- [11] O'Rourke, J.: Finding minimal enclosing boxes. In: International journal of computer and information sciences 14(3), 183-99, 1985
- [12] Rank, E.; Wunderlich, T.; Schäfer, T.; Schleinkofer, M.: Abschlussbericht BayFORREST F248. Vom Laseraufmaß zum Stoffflussmanagement für Altbauten. München, Bayerisches Staatsministerium für Umwelt, Gesundheit und Verbraucherschutz, 2005, <http://www.inf.bauwesen.tu-muenchen.de/forschung/bayforrest/f248/dokumente/F248%20-%20Abschlussbericht.pdf>, Stand 06.02.2006
- [13] Rumelhart, D.; Hinton, G; Williams, R.: Learning representations by back-propagating errors., Nature 323, S. 533-536, London 1986
- [14] Rojas, R.: Neural networks - A systematic introduction. Springer-Verlag, Berlin, New York 1996
- [15] Rosenblatt, F.: The Perceptron. A Probabilistic Model for Information Storage and Organization in the Brain. In: Cornell Aeronautical Laboratory (Hrsg.): Psychological Review, v65, No. 6, pp. 386-408, 1958
- [16] Schleinkofer, M.; Schäfer, T.; Rank, E.: BayFORREST F248. Vom Laseraufmaß zum Produktmodell für Altbauten. <http://www.inf.bauwesen.tu-muenchen.de/forschung/bayforrest/f248>, Stand 01.02.2006
- [17] Schleinkofer, M.; Bastian, A.; van Treeck, C.; Rank, E.: Automatic Component Identification using Artificial Neural Network Techniques. In: Topping, B. (Hrsg.): Proceedings of the Eighth International Conference on the Application of Artificial Intelligence to Civil, Structural and Environmental Engineering, Rome, Italy: Civil-Comp Press 2005
- [18] Schwarz, H.: Numerische Mathematik, Teubner, Stuttgart 1997
- [19] Stöcker, H. (Hrsg.): Taschenbuch mathematischer Formeln und moderner Verfahren, Deutsch, Thun 1999
- [20] Thurow, T.: Digitaler Architekturbestand. Untersuchungen zur computergestützten, schrittweisen Erfassung und Abbildung der Geometrie von Gebäuden im Kontext der planungsrelevanten Bauaufnahme. Dissertation, Bauhaus-Universität, Weimar, 2004
- [21] van Treeck, C.: Gebäudemodell-basierte Simulation von Raumlufströmungen. Dissertation, Technische Universität München, Shaker 2004
- [22] van Treeck, C.; Rank, E.: Dimensional reduction of 3D building models using graph theory and its application in building energy simulation, Engineering with Computers, 2006
- [23] Welstead S.: Neural Network and fuzzy logic applications in c/c++, John Wiley & Sons, New York 1994
- [24] Wikipedia (Hrsg.): Alter Hof, http://de.wikipedia.org/wiki/Alter_Hof, Stand 08.05.2006

- [25] Wilke, W.: Segmentierung und Approximation großer Punktwolken. Dissertation, Technische Universität Darmstadt, Darmstadt 2002
- [26] Wunderlich, T.: Terrestrial Laser Scanning - an important Step towards Construction Information. In: Proc. of the 26th FIG General Assembly, Paris, 2003
- [27] Zell, A.: Simulation neuronaler Netze, Oldenbourg, München 2000