

Zur maßstabsabhängigen Erzeugung von 3D-Stadtmodellen

Von der Fakultät Luft- und Raumfahrttechnik und Geodäsie
der Universität Stuttgart zur Erlangung der Würde eines Doktors
der Ingenieurwissenschaften (Dr.-Ing.) genehmigte Abhandlung

Vorgelegt von
Martin Kada
aus Rendsburg

Hauptberichter:	Prof. Dr.-Ing. habil. Dieter Fritsch
Mitberichter:	Prof. Dr. rer. nat. Thomas Ertl
Mitberichterin:	Prof. Dr.-Ing. habil. Monika Sester

Tag der mündlichen Prüfung: 27. Juli 2007

Institut für Photogrammetrie
der Universität Stuttgart

2007

Inhaltsverzeichnis

Zusammenfassung	4
Summary	6
1 Einleitung	8
1.1 Motivation	8
1.2 Zielsetzung	10
1.3 Gliederung	11
2 3D-Stadtmodelle	12
2.1 Festkörpermodellierung	12
2.1.1 Dekompositionsmodelle	12
2.1.2 Konstruktive Modelle	14
2.1.3 Randbeschreibung	15
2.2 3D-Gebäudemodelle	16
2.3 Gebäudeextraktion	17
2.4 Fassadenrekonstruktion	21
2.5 CityGML	23
3 Generalisierung	25
3.1 Generalisierung von 2D-Gebäudegrundrissen	26
3.2 Generalisierung von 3D-Gebäudemodellen	31
3.3 Flächensimplifizierung	37
3.3.1 Vertex Clustering	38
3.3.2 Vertex Decimation	39
3.3.3 Edge Collapse	40
3.3.4 Progressive Netze	41
3.3.5 Voxel-Basierte Simplifizierung	41
3.3.6 Bild-Basierte Simplifizierung	41
3.3.7 Simplifizierung mittels Alpha-Shapes	42
3.3.8 Erscheinungsbild bewahrende Simplifizierung	42
3.3.9 Erkennung und Entfernung von Merkmalen	42
4 3D-Generalisierung mittels Flächensimplifizierung	45
4.1 Die iterative Generalisierung	45
4.2 Der Relationsgraph	46
4.3 Formvereinfachung	49
4.3.1 Erkennung und Entfernung von Merkmalen	50
4.3.2 Innere Punkte	51

4.4	Formeinpassung	52
4.5	Texturen	55
4.6	Ergebnisse	56
4.7	Diskussion	58
5	Generalisierung zur globalen Formvereinfachung	60
5.1	Generalisierung mittels Zellenzerlegung	60
5.2	Ableitung von Teilungsebenen	63
5.2.1	Selektion der Eingabepolygone	63
5.2.2	Pufferbildung	63
5.2.4	Ausgleichung der Teilungsebenen	66
5.3	Erzeugung der 2.5D-Zellenzerlegung	66
5.4	Identifikation der Gebäudezellen	67
5.5	Generierung der 3D-Dachform	68
5.6	Optimierung der Primitivauswahl	70
5.6.1	Auffüllen von Löchern	70
5.6.2	Entfernen von Primitiven	70
5.6.3	Zusammenziehen von Knoten	71
5.7	Ergebnisse	72
5.8	Alternativer Ansatz zur Generierung der 3D-Dachform	76
6	Fassadenausgestaltung	79
6.1	Manuelle Texturierung von Gebäudefassaden	79
6.2	Automatische Texturierung von Gebäudefassaden	81
6.2.1	Korrektur der Linsenverzeichnung	81
6.2.2	Texturextraktion und Platzierung	83
6.2.3	Erkennung von Selbstüberdeckungen	84
6.2.4	Bildfusion	85
6.2.5	Realisierung mit einer Grafikipipeline	86
6.2.6	Bewertung	87
6.3	Fassadenrekonstruktion	89
6.3.1	Vorverarbeitung	90
6.3.2	Segmentierung von Fensterkanten	91
6.3.3	Zellenzerlegung	92
6.3.4	Grobklassifizierung	93
6.3.5	Feinklassifizierung	95
6.3.6	Feinmodellierung mittels Hypothesentests	95
7	Diskussion und Ausblick	98
8	Literaturverzeichnis	101
	Lebenslauf	111

Zusammenfassung

Die großflächige Erfassung von 3D-Stadtmodellen erfolgt bislang durch die (semi-) automatische Interpretation von Luftbildern und Luftlaserdaten. Aufgrund der luftbasierten Auswertung besitzen die resultierenden Gebäudemodelle zwar detaillierte Grundrisse und Dachstrukturen; eine Rekonstruktion der Fassaden ist auf dieser Datengrundlage jedoch nicht möglich. Folglich werden die Grundrisse bei der Modellierung bis zur gemessenen Dachkante einfach extrudiert. Für viele Anwendungen ist dieser Detaillierungsgrad mehr als ausreichend. Werden dennoch 3D-Gebäudemodelle mit detaillierten Fassaden benötigt, können terrestrische Bild- und Laserdaten Abhilfe schaffen. Deren Interpretation ist allerdings zeitaufwendig, da der hierfür notwendige Automationsgrad noch nicht erreicht ist. Hinzu kommt, dass die terrestrische Datenerfassung meist nur unvollständig erfolgen kann. Gründe hierfür sind unvermeidliche Überdeckungen der zu messenden Fassadenfläche durch Fremdobjekte, sowie der Umstand, dass Gebäudeseiten mit dem Messinstrument nicht erreichbar sind. Eine Integration von luftgestützt erfasster und terrestrischer Daten könnte hierfür eine Lösung darstellen.

Neben den klassischen Anwendungen von 3D-Stadtmodellen, wie z.B. der Standortplanung von Mobilfunkantennen, Ausrichtung von Solaranlagen und Lärmanalysen, nimmt die Visualisierung einen immer höheren Stellenwert ein. Hierfür werden sowohl großflächige 3D-Gebäudedaten mit geringer Detaillierung als auch punktuell photorealistische Modelle benötigt. In der Echtzeit- und webbasierten Visualisierung werden Objekte in Abhängigkeit von der Entfernung zum Betrachter unterschiedlich detailliert dargestellt. Kartenverwandte 3D-Darstellungen betonen zur besseren Orientierung wichtige Landmarken, müssen dabei aber graphische Mindestgrößen einhalten. Für nicht-photorealistische Visualisierungen werden Modelle mit einer möglichst geringen Anzahl von Kanten benötigt, welche trotzdem die charakteristischen Gebäudeformen widerspiegeln.

Es werden folglich, je nach Anwendungsgebiet, 3D-Gebäudemodelle in unterschiedlichen Detaillierungsgraden nachgefragt. Da deren Erfassung mit hohen Kosten verbunden ist, wird eine breite Verwendung und kommerzielle Nutzung dieser Daten angestrebt. Um den Austausch zu vereinfachen, wurde von der Special Interest Group 3D (SIG 3D) der Initiative Geodateninfrastruktur Nordrhein-Westfalen (GDI NRW) das Anwendungsschema CityGML entwickelt und zur Standardisierung dem Open Geospatial Consortium (OGC) vorgelegt. Drei der darin enthaltenen fünf Detailstufen unterscheiden Gebäudemodelle aufgrund ihrer Außenhülle in Blockgebäude, Modelle mit Dachstruktur und Architekturmodelle. Dessen ungeachtet ist es von der Datenerfassung und Datenhaltung her nicht sinnvoll, die Modelle in allen erdenklichen Detailstufen zu rekonstruieren und auf lange Zeit vorzuhalten. Vielmehr sollten sie in der durch die Erfassungsmethode bedingten Detaillierung verwaltet und erst bei Bedarf automatisch und maßstabsgerecht in andere Detaillierungsgrade umgewandelt werden.

Die vorliegende Arbeit soll hierzu einen Beitrag leisten, indem Lösungen in Form der kartographischen Vereinfachung von Modellen mit Dachstrukturen und der bildlichen und geometrischen Ausgestaltung von Fassaden vorgestellt werden.

Bevor konkrete Lösungen präsentiert werden, erfolgt in der Arbeit zunächst eine allgemeine Diskussion zur Modellierung und Rekonstruktion von 3D-Stadtmodellen. Da Gebäude hauptsächlich aus ebenen Flächen bestehen, wird diese Objektart ausschließlich als Festkörper modelliert. Hierbei dominieren die Modellierungsformen der Parameterdarstellung, Konstruktion mit Raumprimitiven und Randbeschreibung. Es existieren zwar eine Reihe weiterer Formen, die sich jedoch speziell für die Modellierung von Gebäudeobjekten nicht gut eignen.

Die geometrische Vereinfachung unter Wahrung der Objektcharakteristika ist ein elementarer Vorgang in der kartographischen Generalisierung. Für 2D-Gebäudedaten existiert bereits eine Reihe von Verfahren, um Grundrisse unter Beibehaltung der Koplanarität, Parallelität und Rechtwinkligkeit von Gebäudeseiten zu generalisieren. Einige davon, darunter die Generalisierung mittels Maßstabsräumen, Musterformen, expliziten Regeln und der Merkmalerkennung, wurden auch schon für die Verarbeitung von 3D-Gebäudemodellen umgesetzt. Jedoch lassen sich die Konzepte nicht so einfach auf den dreidimensionalen Fall übertragen. Bei-

spielsweise sind Dachstrukturen häufig nicht rechtwinklig, so dass diese Verfahren oftmals Schwierigkeiten mit deren Vereinfachung haben. Ähnliches gilt für die Simplifizierungsverfahren aus der Computergrafik, die für allgemeine Flächen entwickelt wurden und demzufolge objektspezifische Charakteristiken nicht bewahren. Da die Konzepte und Methoden der beschriebenen Forschungsgebiete direkten Einfluss auf die Entwicklung der in der Arbeit dokumentierten Generalisierungsverfahren hatten, werden auch sie ausführlich diskutiert.

Auf diesen Grundlagen aufbauend werden zwei neue Verfahren zur Generalisierung bzw. geometrischen Vereinfachung von einzelnen 3D-Gebäudemodellen vorgestellt. Diese sind von ihrer Vorgehensweise komplementär.

Das erste Verfahren arbeitet von unten nach oben gerichtet (bottom up), indem die Merkmale von Gebäuden in einem iterativen Prozess erkannt und entfernt werden. Bei einem Merkmal kann es sich beispielsweise um eine Ausbuchtung an der Fassade handeln. Die hierzu notwendigen und teilweise komplexen Operatoren werden durch die Kombination elementarer Flächensimplifizierungsoperatoren gebildet. Sofern vorhanden, sollen diese Operatoren die schon erwähnten geometrischen Relationen Koplanarität, Parallelität und Rechtwinkligkeit zwischen den Flächen strikt erhalten. Dies wird zusätzlich mittels des neu eingeführten Relationsgraphen sichergestellt, welcher die Relationen des Gebäudemodells verwaltet und anhand dessen die Gültigkeit jeder Operation überprüft werden kann. Nach der Geometrievereinfachung erfolgt eine Einpassung der neuen Gebäudeform an das Originalmodell. Hierzu wird das Gebäudemodell in ein funktionales Modell konvertiert und einer Ausgleichung nach kleinsten Quadraten unterzogen. Dadurch werden eventuelle Veränderungen in der Lage und den Proportionen wieder korrigiert, so dass das vereinfachte 3D-Gebäudemodell optimal den Raum des Ursprungsmodells einnimmt.

Während in diesem Ansatz auf der Detailebene gearbeitet wird, um kleine Strukturelemente zu eliminieren, erfolgt die Generalisierung im zweiten Verfahren von oben nach unten gerichtet (top down), durch einen Nachbau der globalen Gebäudeform. Hierfür werden aus den polygonalen Fassadenflächen eine Menge approximierender Ebenen abgeleitet, um aus diesen eine generalisierte 2.5D-Zellenzerlegung des Grundrisses zu erhalten. Anschließend wird für jede Zelle eine vereinfachte Dachform erzeugt und die resultierenden 3D-Primitive zu einem 3D-Gebäudemodell zusammengefügt. Durch die Interpretation der Dachform ist nicht nur eine Vereinfachung möglich, sondern es wird auch gezeigt, wie der elementare Generalisierungsvorgang Typifizieren für Dachformen, realisiert werden kann. Dabei handelt es sich um die Ausdünnung einer Menge gleichartiger Objekte unter Beibehaltung der ursprünglichen Objektverteilung.

Neben der Generalisierung ist die Ausgestaltung von Fassaden eine weitere Aufgabe, die zur Bereitstellung von Modellen in unterschiedlichen Detailstufen, gelöst werden müssen. Beginnend mit einer Diskussion zur manuellen Texturierung wird darauf aufbauend ein automatischer Ansatz entwickelt. Dabei werden aus Bildern, deren äußere Orientierungen zum Aufnahmezeitpunkt bekannt sind, Fassadenbilder extrahiert und auf die Fassadengeometrie aufgebracht. Überdeckungen durch das Objekt selbst oder durch Fremdkörper werden erkannt und über die Fusion mehrerer Bilder retuschiert. Es wird gezeigt, dass sich der Prozess unter Ausnutzung spezieller Grafikhardware realisieren lässt, um beispielsweise bei der mobilen Erfassung die Ergebnisse in Echtzeit darzustellen. Nicht in allen Situationen reichen Fassadenbilder für eine adäquate Visualisierung aus; oder es wird für Analyseaufgaben eine semantische Strukturierung von der Fassade benötigt. Zu diesem Zweck wird eine Interpretation von 3D-Punktwolken vorgestellt, die mittels eines terrestrischen Laserscanners erfasst wurden. Fenster und Türen werden über eine Segmentierung und einfach gestalteter Regeln detektiert und ausmodelliert. Wie bei der Generalisierung erfolgt die Modellierung per Zellenzerlegung.

Die Ergebnisse zur Generalisierung und der Ausgestaltung von Fassaden, die im Laufe der Untersuchungen entstanden sind, werden an den jeweiligen Stellen aufgeführt und ausführlich erörtert. Eine Diskussion der Verfahren und ein Ausblick schließen die Arbeit ab.

Summary

Large area 3D city models have been generated so far by (semi-) automatic interpretation of aerial images and laser data. Because of the utilisation of aerial data the resulting building models have detailed ground plans and roof structures. However, a reconstruction of the façade is not possible based on this kind of data. Therefore, the ground plan line segments are simply extruded in the modelling process to the measured edges of the roof. This level of detail is sufficient for many applications. If 3D building models with detailed façades are needed, then terrestrial images and laser data must be used. Their interpretation is very time consuming as the required degree of automation has not yet been reached. In addition, the terrestrial data collection is in most cases incomplete. The reasons are inevitable occlusions of the measured façade surface by other objects and circumstances where building sides may not be accessible with the measuring device. An integration of aerial and terrestrial data may be a solution for this.

Besides the traditional applications of 3D city models, like e.g. the planning of mobile antennas, alignment of solar installations and noise analyses, the visualisation of urban areas gains in importance. For this purpose, 3D building data is needed for large areas in different levels of detail. In real-time and web-based visualisation, object details are shown or omitted depending on their distances to the viewer. Map-like 3D presentations emphasises important landmarks for orientation purposes, but have to maintain cartographic minimal distances between the objects. For non-photorealistic visualisations, models with preferably the lowest number of edges are required that still reflect the characteristic shapes of the buildings.

Because the acquisition of 3D building models is very expensive, it is necessary to support a broad variety of commercial applications. In order to ease data exchange, the application schema CityGML has been developed by the Special Interest Group 3D (SIG 3D) of the initiative Geodata Infrastructure North-Rhine Westphalia (GDI NRW) and presented to the Open Geospatial Consortium (OGC) for standardisation. Three of the five levels of detail distinguish building models by their outside appearance: block models, models with a roof structure and architectural models. Though, from the data acquisition and storage point of view, it is not reasonable to reconstruct and maintain models in all imaginable levels of detail. Rather, the models should be stored in their original level of detail and automatically transformed to scale on demand.

The work at hand ought to make a contribution by proposing solutions for the cartographic simplification of models with roof structures and the reconstruction of façades with images and real geometry.

Before the methods and techniques are presented, a general discussion of the modelling and reconstruction of 3D city models takes place. As buildings are constructed as planar surfaces, these kinds of objects are exclusively generated as solid models. Here, the dominant concepts are primitive instancing, constructive solid geometry and boundary representations. There are also other representations which, however, are not very well suited for modelling building objects.

The geometric simplification which protects the object characteristics is an elemental operation in the cartographic generalisation. There exist a number of generalisation methods for 2D building ground plans that preserves the coplanarity, parallelism and rectangularity of line segments. Some of them, like the generalisation based on scale spaces, templates, explicit rule sets and feature detection and removal, were modified to work with 3D building models. However, the concepts can not be easily made to work with 3D data. E.g. do the roof structures of 3D building models rarely have right angles, which these approaches have difficulties with simplifying. Same goes for the surface simplification methods from computer graphics, which have been developed for general shapes and therefore can not preserve the specific characteristics of buildings. As the concepts and methods of the aforementioned research fields influenced the development of the described approaches, they are discussed in detail within this study.

Based on these principles, two new approaches for the generalisation of single 3D building models are presented. Both are complementary in their course of action.

The first approach works bottom up by detecting and removing features in an iterative process. A feature can e.g. be an extrusion on the façade. As already mentioned, these operators must strictly preserve the geometric

relations coplanarity, parallelism and rectangularity of faces. The necessary operators are therefore built by combining several elementary operations. The relations are additionally guaranteed by the introduction of a relations graph. Relations are managed by this graph, which the generalisation process uses to verify the validity of each operation. After the simplification of the geometry, the new building shape is fit to the original model. For this purpose, the building model is converted into a functional model and its parameters optimised in a least squares adjustment. Thereby, correcting any changes in position and the proportions of the faces, so that the simplified 3D building model best fits into the original space.

While the former process works on the detail level to eliminate small structural elements, the generalisation in the second approach works top down by rebuilding the model in its global shape. Therefore, a set of approximating planes are derived from the polygonal faces in order to generate a 2.5D cell decomposition of the ground plan. Afterwards, the roof shape is build for each cell and the resulting 3D primitives are merged together. By doing an interpretation of the roof structure it is not only possible to do a simplification, but also the generalisation operator typification can be implemented for some roofs types. A number of equally shaped objects are replaced by a lower number while maintaining the object arrangement.

Another topic besides the generalisation is the reconstruction of façades. Starting with a discussion of manual texturing, an automatic approach is build that generates façade images. This is done from terrestrial images, for which the external orientation is known for the time of their acquisition. Façade images are extracted and mapped onto the façade geometry. Occlusions from the object itself and other objects are detected by this approach and retouched by image fusion. It is shown, that the process can be applied to special graphics hardware for the efficient capturing of data in real-time. Not in all situations, façade textures are adequate for visualisation or the semantic structure of the façade is necessary. For this purpose, an interpretation of 3D point clouds from terrestrial laser scanning is presented. Windows and doors are detected and modelled by a segmentation process and a set of simple rules. As in the generalisation the modelling is done using cell decomposition.

The results of the generalisation and reconstruction of façades, which were gathered during the study, are presented and discussed in detail at the respective chapters. A discussion and an outlook conclude the work.

1 Einleitung

1.1 Motivation

Die Erfassung und Präsentation von 3D-Stadtmodellen ist seit über 15 Jahren Thema verschiedener wissenschaftlicher Disziplinen und ist auch heute noch ein äußerst aktives Forschungsgebiet. Unter einem dreidimensionalen Stadtmodell versteht man im Allgemeinen das digitale Abbild von Geländeoberfläche und den 3D-Gebäuden sowie verstärkt auch Straßen, Vegetation und Stadtmöblierung eines urbanen Gebiets. Mittlerweile existieren einige kommerzielle Softwaresysteme und Dienstleistungsunternehmen, welche Gebäude in unterschiedlicher geometrischer Detaillierung rekonstruieren. Der Reichtum an geometrischen Details variiert in Abhängigkeit von der Erfassungsmethode. Eine großflächige Erfassung von 3D-Gebäudemodellen, wie sie von Städten und Kommunen häufig in Auftrag gegeben werden, ist nur durch die Auswertung von Luftbildern und Luftlaserdaten möglich. Aufgrund der luftbasierten Datenerfassung sind gering- bis mitteldetaillierte Modelle mit flachen Fassaden und einfachen Dachstrukturen möglich. Die Rekonstruktion detaillierter Fassadenstrukturen kann nur aus terrestrischen Bild- und Geometriedaten erfolgen, deren Modellierung aufgrund des bisher erreichbaren Automationsgrads äußerst zeitaufwendig ist. Des Weiteren kann häufig die Datenerfassung nur unvollständig erfolgen. Gründe dafür sind unvermeidliche Verdeckungen der Fassadenfläche durch Fremdobjekte und nicht öffentlich zugängliche oder mit dem Messinstrument nur schwer erreichbare Gebäudeseiten. In der Praxis erfolgt eine detaillierte Modellierung daher fast ausschließlich für die zur Straße zugewandten Gebädefassaden oder für vereinzelte Landmarken und Wahrzeichen. Zur Lösung dieses Problems wird derzeit die Integration luftgestützt erfasster und terrestrischer Daten für die automatische Modellierung von Gebäudeobjekten (z.B. [Früh und Zakhor 2003]) sowie die Erzeugung synthetischer Datensätze für nicht sichtbare Fassadenbereiche intensiv erforscht (z.B. [Pauly et al. 2005], [Mitra et al. 2006]).

Zu den klassischen Hauptanwendungen von 3D-Stadtmodellen gehört die Planung von Antennenstandorten für Mobilfunknetze. Basierend auf den Dachformen und der exakten Berechnung zur Ausbreitung von Funkwellen lässt sich bestimmen, welche Gebäude für die Positionierung von Sendestationen am besten geeignet sind. Energieversorger benötigen aktuelle 3D-Stadtmodelle, um Solaranlagen optimal auszurichten. Eine EU-Richtlinie aus dem Jahr 2002 zur Minderung von Umgebungslärm fordert genaue Angaben zur Lärmbelastung, die nur mithilfe von 3D-Modellen ableitbar sind. Neben den Analysen ist die hauptsächliche Nutzung von 3D-Stadtmodellen die Visualisierung (vgl. [Schilling et al. 2004], [Bauer und Mohl 2004]). Hierfür wünschen Stadtplaner und Marketingabteilungen sowohl großflächig 3D-Gebäudedaten mit geringer Detaillierung als auch punktuell fotorealistische Modelle. Kommende Fahrzeugnavigationssysteme werden den Fahrer dreidimensional durch die urbane Umgebung zum Ziel führen. Zum Aufbau realistischer Flug-, Seefahrts- und Fahrsimulationssysteme werden sowohl einfache Gebäudemodelle mit Dachstrukturen und Texturen als auch höher detaillierte Gebäude mit Fassadenstrukturen (Türen, Fenster, Balkone, etc.) benötigt [Bildstein 2005]. Eine zunehmende Bedeutung erfahren 3D-Stadtmodelle auch in der Tourismusbranche, um Städte oder Regionen zu fördern. Der virtuelle Globus Google Earth integriert 3D-Gebäudemodelle, um lokationsabhängig zu werben und Informationen über das Internet auszutauschen. Eine umfassende Auflistung weiterer Anwendungen und deren spezifischen Anforderungen an 3D-Stadtmodelle ist in [Albert et al. 2003] gegeben.

Je nach Anwendungsgebiet werden 3D-Gebäudemodelle folglich mit unterschiedlichen Detaillierungsgraden nachgefragt. Vor allem bei der Echtzeit- und webbasierten Visualisierung ist es sinnvoll die Modelle gleich in mehreren Detailstufen verfügbar zu haben um Objekte in Abhängigkeit von der Entfernung zum Betrachter unterschiedlich detailliert darzustellen. Bei dieser als Level of Detail (LOD) bezeichneten Technik kommen im Vordergrund Modelle in der höchsten Detailstufe und mit zunehmender Entfernung Modelle in niedrigeren Detailstufen zum Einsatz. Dadurch nimmt die Komplexität der Geometrie- und Bilddaten einer Szene ab und eine effiziente Übertragung und Visualisierung derselben ist möglich. Gewöhnlich findet eine Aufteilung in den drei diskreten Detailstufen *Klötzchenmodell*, *einfaches Gebäude mit Dachstruktur* und *Architekturmodell* mit detaillierten Fassadengeometrien statt. Diese Einteilung ist durch die Erfassungsmethoden

bzw. Konstruktionsstrategien motiviert, besitzt aber ansonsten keinerlei qualitative oder quantitative Aussagekraft [Gröger et al. 2004]. Die Arbeitsgruppe SIG3D der Initiative Geodateninfrastruktur NRW schlägt einen Standard für den Austausch und die Speicherung von 3D-Stadtmodellen vor. In der Spezifikation des unter dem Namen CityGML getauften Standards sind zwei zusätzliche Detailstufen für die Repräsentation großräumiger Regionalmodelle und für die Modellierung von Innenräumen vorgesehen. Des Weiteren werden für alle Detailstufen absolute Genauigkeiten und Minimaldimensionen definiert. Die Spezifikation liegt mittlerweile dem technischen Gremium des Open Geospatial Consortium (OGC) als Diskussionsgrundlage für einen internationalen Standard vor [Gröger et al. 2006].

Im Bereich der ortsbezogenen Dienste und der kontextbezogenen Anwendungen ist eine orts- bzw. situationgerechte Präsentation der für die zu erfüllende Aufgabe des Benutzers relevanten Umgebungsinformationen von enormer Wichtigkeit. Um jederzeit und überall nutzbar zu sein, laufen diese Systeme üblicherweise auf mobilen Endgeräten wie persönliche digitale Assistenten (PDA) oder Mobiltelefonen. Deren Ausgabefläche ist limitiert durch die vom Benutzer maximal akzeptierten Ausmaße des Endgeräts und wird sich daher in zukünftigen Hardwaregenerationen auch kaum noch ändern. Bedingt durch die geringe Auflösung der Bildschirme im Vergleich zu Papierkarten gelten größere Werte für die graphischen Mindestgrößen (vgl. [Neudeck 2001]). Diese Werte sind darüber hinaus umso höher anzusetzen, je geringer der Kontrast ist und je feiner die Farbabstufungen sind [Hake et al. 2002]. Folglich lässt sich bei der digitalen Darstellung von Karten eine geringere Informationsdichte als in der Papierform beobachten. Um im Fokusbereich des Betrachters dennoch alle relevanten Details einer zweidimensionalen Karte lesbar darzustellen, schlagen Harrie et al. [2002] eine vektorbasierte Fischaugen-Ansicht vor, bei der die Gebäudegrundrisse in Echtzeit geometrisch vereinfacht werden. Die Detaillierung der Grundrisse nimmt ab je weiter entfernt sich diese vom Fokuszentrum befinden. Da 3D-Gebäudemodelle an und für sich schon komplexer als 2D-Daten sind, ist eine geometrische Vereinfachung von detaillierten Objekten in jedem Fall notwendig, um die graphischen Mindestgrößen einzuhalten. Dies betrifft vor allem Gebäudemodelle von Landmarken, da sie eine markante und damit auch häufig komplexe äußere Form oder Fassadengestaltung besitzen. Eine geometrische Vereinfachung darf jedoch nur unwichtige Details entfernen und muss die für das Objekt wichtigen bzw. charakteristischen Merkmale unter allen Umständen erhalten. Bei dem vereinfachten Gebäudemodell einer Kirche darf beispielsweise der Kirchturm nicht fehlen, da das Objekt ansonsten nur noch schwer als Kirche zu erkennen ist. Aber auch Gebäudemodelle mit geringer Bedeutung müssen für die kartenverwandte Darstellung geeignet aufbereitet werden um den Blick des Betrachters nicht unnötig auf sich zu ziehen. In einigen Fällen verzichtet man sogar vollständig auf eine dreidimensionale Präsentation der unwichtigen Gebäude und nur die Landmarken werden als 3D-Gebäude dargestellt. Für Anwendungen auf mobilen Endgeräten bedeutet diese geometrische Vereinfachung ein geringeres Volumen bei der Übertragung und Haltung der Daten und damit auch eine Beschleunigung der Kartendarstellung. Außerdem ist aufgrund der geringeren geometrischen Komplexität die dargestellte Situation besser erkennbar und leichter für den Betrachter zu interpretieren.

Eine photorealistische Darstellung ist nicht in jedem Fall geeignet um eine räumliche Situation wie beabsichtigt zu kommunizieren. Im Bereich der Architektur und Design werden beispielsweise computergenerierte Bilder von CAD-Modellen oftmals mit einem Zeichenstift per Hand auf Papier durchgepaust. Das skizzenähnliche Abbild soll das Objekt lebendiger wirken lassen und der Präsentation den Charakter eines vorläufigen Entwurfs vermitteln. Um die Zeichengenauigkeit zu simulieren, lassen Van Bakergem und Obata [1991] den Stift eines Plotters während der Ausgabe wackeln, wodurch ein ähnliches Ergebnis wie bei einer Freihandzeichnung erreicht wird. Strothotte et al. [1994] stellen einen softwarebasierten Ansatz vor, um skizzenhafte Darstellungen von allgemeinen Objekten zu erhalten. Durch die Verwendung verschiedenartiger Linienstile (Dicke, Sättigung, Krümmung, etc.) wird der Blick des Betrachters auf bestimmte Regionen der dargestellten Szene gelenkt. Diese als nicht-photorealistisches Rendering bezeichnete Darstellungsform wird von Buchholz et al. [2005] genutzt, um interaktive 3D-Stadtmodelle im Hinblick auf die Wahrnehmungsfähigkeiten des Menschen hin zu optimieren. Demgemäß sollen eine effektive Kommunikation der Struktur und der Beziehungen urbaner Objekte sowie deren thematische Informationen möglich sein. Dadurch ist auch bei einer für die photorealistische Visualisierung nicht ausreichenden Detaillierung von Geometrie und Bildmaterial eine hochqualitative Präsentation möglich. Eine der genutzten Techniken ist die in Döllner und Walther [2003] vorgestellte skizzenähnliche Betonung von Gebäudekanten mittels stilisierter Striche. Für 3D-Gebäudemodelle mit vielen kleinen Ein- und Ausbuchtungen führt dies zu einer Vielzahl harter Kanten, die nicht charakteristisch für das Objekt sind und bei zu hoher Dichte zu Bildstörungen führt. Daher ist im Vorfeld der Visualisierung eine Detektion der wichtigen Gebäudekanten notwendig. Alternativ ist die Anwendung eines Generalisierungsschemas angedacht, welches die 3D-Gebäudemodelle mit kartographischen

Methoden geometrisch vereinfacht. Das Ergebnis soll ein Modell mit einer möglichst geringen Anzahl von Kanten sein, welches die charakteristische Gebäudeform widerspiegelt.

1.2 Zielsetzung

Vom Gesichtspunkt der Datenerfassung und Datenhaltung ist es nicht sinnvoll, 3D-Stadtmodelle in allen erdenklichen Detailstufen zu rekonstruieren und auf lange Zeit vorzuhalten. Jede Anwendung hat individuelle Anforderungen an Genauigkeit und graphischen Mindestgrößen, die zum Zeitpunkt der Erfassung nicht bekannt sein müssen. Vielmehr werden 3D-Stadtmodelle in dem durch die Erfassungsmethode bedingten Detaillierungsgrad verwaltet und sind bei Bedarf automatisch in die nachgefragte Detailstufe umzuwandeln. Dabei ist zu beachten, dass für 3D-Gebäudemodelle spezielle Formeigenschaften gelten. Deren Flächen sind überwiegend planar und Fassadenelemente besitzen häufig eine koplanare, parallele oder orthogonale Anordnung. Diese allgemeinen Gebäudeeigenschaften sowie die charakteristische Gebäudeform müssen bei der Transformation strikt erhalten bleiben. Eine geometrische Vereinfachung unter den genannten Restriktionen wird als Generalisierung bezeichnet.

Ziel der vorliegenden Arbeit ist die Entwicklung von Verfahren und Methoden zur automatischen Erzeugung von 3D-Gebäudemodellen in unterschiedlichen Detailstufen. Da 3D-Stadtmodelle, die aus luftgestützt erfassten Daten rekonstruiert werden, mittlerweile eine hohe Qualität besitzen, kann auf diese Datengrundlage zurückgegriffen werden. Im Speziellen stand für die Forschungsdauer das 3D-Stadtmodell von Stuttgart zur Verfügung, welches freundlicherweise vom Stadtmessungsamt Stuttgart bereitgestellt wurde. Dabei handelt es sich um einen typischen Datensatz, den Städte und Kommunen in ihrem Bestand besitzen. Wie schon eingangs erwähnt, besitzt diese Art von Modellen aufgrund der eingesetzten Erfassungsmethode (siehe [Wolf 1999]) sehr detaillierte Grundrisse und Dachstrukturen, jedoch nur flache Fassadenelemente ohne Fenster, Türen, Balkone, etc.. Die sehr detaillierten Grundrissdaten stammen dabei aus der Automatisierten Liegenschaftskarte (ALK), die als Basisdaten bei der Gebäuderekonstruktion eingesetzt wurden. Die Arbeit zeigt sowohl die Generalisierung einzelner 3D-Gebäudemodelle bis hin zu Blockmodellen als auch die bildliche und geometrische Ausgestaltung von Fassaden (siehe Abbildung 1-1). Es wird somit die technische Grundlage geschaffen, um 3D-Gebäudemodelle in beide Richtungen der Detaillierungsskala hin zu transformieren und damit die Anforderungen eines möglichst breiten Spektrums von Anwendungsfeldern abzudecken.

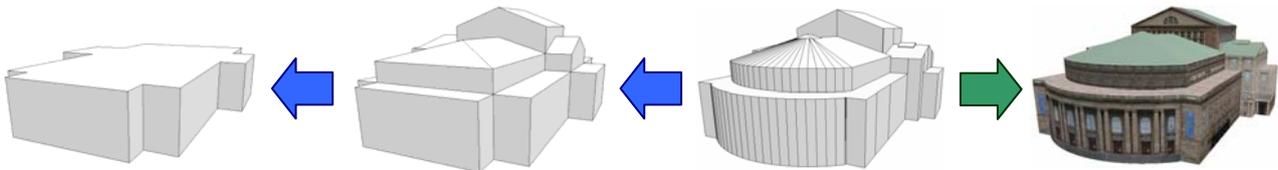


Abbildung 1-1: Geometrische Vereinfachung (blau) eines luftgestützt erfassten 3D-Gebäudemodells sowie die bildliche und geometrische Ausgestaltung der Gebäudefassade (grün).

Im Themenbereich der Generalisierung von 3D-Gebäudemodellen werden zwei sehr gegensätzliche Ansätze entwickelt. Der Erste fundiert auf der klassischen Idee von Staufenbiel [1973] zur Generalisierung von 2D-Gebäudegrundrissen. Die Vereinfachung von Modellen geschieht iterativ mithilfe eines expliziten Regelwerks, welches geometrische Situationen und darauf abgestimmte Generalisierungsmaßnahmen beschreibt. Eine Erweiterung des Ansatzes auf die Verarbeitung dreidimensionaler Daten, welche in Randbeschreibung gegeben sind, erfordert neue, angepasste Regeln, die in der vorliegenden Arbeit mit den aus der Computergrafik bekannten Operatoren der Flächensimplifizierung realisiert sind. Die Regeln werden sukzessiv angewendet bis der gewünschte Detaillierungsgrad erreicht ist. Um anschließend die allgemeinen Gebäudeeigenschaften der vereinfachten Modelle zu erzwingen, werden wie schon in [Sester 2000] die Gebäudemodelle einer Ausgleichung nach kleinsten Quadraten unterzogen. Zu diesem Zweck wird ein funktionales Modell aufgestellt, welches die polygonalen Flächen der 3D-Gebäudemodelle koplanar, parallel und orthogonal zueinander einregelt. Außerdem werden die Eckpunkte der ursprünglichen Geometrie mit einbezogen, so dass die vereinfachte Gebäudeform durch die Ausgleichung bestmöglich in die ursprüngliche Form angepasst wird. Dabei werden auch die durch die Vereinfachung entstandenen Verschiebungen und Verzerrungen wieder korrigiert.

Der zweite Ansatz ist von der Vorgehensweise komplementär. Während bisher auf der Detailebene gearbeitet wurde um kleine Strukturelemente zu eliminieren, erfolgt die Generalisierung des 3D-Gebäudemodells durch Nachbau der globalen Gebäudeform. Dabei wird auf die aus der Festkörpermodellierung bekannte Methode der Zellenzerlegung zurückgegriffen. Mittels Zellenzerlegung können beliebig komplex geformte Polyeder als eine Menge einfacher Primitive, auch als Zellen bezeichnet, konstruiert werden. Zellen sind nicht überlappende Polyeder, deren Flächen vollständig an maximal eine Nachbarzelle angrenzen. Der vereinfachte Nachbau des 3D-Gebäudemodells erfolgt ähnlich wie bei der Halbraummodellierung durch Verschneidung des unendlichen Raums entlang vorher definierter Ebenen. Dabei approximieren die Ebenen die ursprüngliche Form des Gebäudemodells. Die vereinfachte Form ergibt sich dann durch Zusammenfügen einer Teilmenge der erzeugten Zellen. Mittels des Ansatzes können sowohl vereinfachte Modelle mit beliebiger Dachstruktur als auch Blockmodelle erzeugt werden. Da Giebel- und Walmdächer die am häufigsten vorkommenden Dachformen sind, wird außerdem die Erweiterung des Ansatzes durch die Anwendung parametrisierter Zellprimitive untersucht.

Nach der Generalisierung von Gebäudemodellen wird anschließend der Frage nachgegangen, wie sich die flachen Gebäudefassaden bildlich und geometrisch ausgestalten lassen. Detaillierte Fassaden sind für alle Anwendungen mit photorealistischer Visualisierung von enormer Bedeutung, insbesondere wenn sich die Betrachter in der Fußgängerperspektive befinden. Generell kann zur Erfüllung diese Aufgabenstellung auf terrestrische Bild- oder Laserdaten zurückgegriffen werden. Beide Datenquellen sind folglich zu betrachten, um automatische Verfahren zur Fassadengestaltung zu entwickeln.

1.3 Gliederung

In den weiterführenden Teilen der vorliegenden Arbeit wird in Kapitel 2 zunächst auf die Modellierung und die Erfassung von 3D-Stadtmodellen eingegangen. Beginnend mit der Festkörpermodellierung werden die wichtigsten Konstruktions- und Repräsentationsformen aufgezeigt. Darauf aufbauend wird ein breiter Überblick über die bis heute bekannten Rekonstruktionsverfahren gegeben. Dabei werden auch die Bezüge zu den verwendeten Ausgangsdaten und Methoden hergestellt, die wiederum ausschlaggebend für die Komplexität und die erreichbare Genauigkeit der rekonstruierten 3D-Modelle sind. Des Weiteren wird außerdem auf die Entwicklung zur Standardisierung von 3D-Stadtmodellen eingegangen, die mit CityGML schon konkrete Formen angenommen hat.

In Kapitel 3 wird der Stand der Forschung im Bereich der geometrischen Vereinfachung einzelner Objekte aufgearbeitet. Da die Generalisierung von grundlegender Bedeutung für die Arbeit ist, wird mit einer allgemeinen Zusammenfassung des Themengebiets angefangen. Anschließend wird eine Zusammenstellung der wichtigsten Arbeiten aus der klassischen 2D-Grundrissgeneralisierung gegeben. Deren Konzepte und Ideen motivieren die meisten Verfahren zur Generalisierung von 3D-Gebäudemodellen. Bevor diese jedoch ausführlich erörtert werden, erfolgt ein Exkurs in das Themengebiet der Flächensimplifizierung. Dabei handelt es sich um leistungsfähige Algorithmen zur Vereinfachung allgemeiner 3D-Objekte. Es werden Verfahren erläutert und der Frage nachgegangen, warum diese sich in ihrer ursprünglichen Form nicht für die Generalisierung von 3D-Gebäudemodellen eignen.

Kapitel 4 stellt ein Generalisierungsverfahren vor, welches die Vorgehensweisen aus der Generalisierung von 2D-Grundrissen mit den Operatoren der Flächensimplifizierung vereint. Am Beispiel realer Gebäudedaten werden Ergebnisse präsentiert und eine Bewertung des Verfahrens vorgenommen. Auch werden die Limitierungen aufgezeigt, die ein solcher Ansatz mit sich bringt. Aus diesen heraus motiviert sich die Generalisierung mittels Zellenzerlegung, welche ausführlich mitsamt Ergebnissen in Kapitel 5 beschrieben ist. Eine Diskussion über den Bezug des Generalisierungsverfahrens zur Rekonstruktion von 3D-Gebäudemodellen rundet das Kapitel ab.

Die Ausgestaltung von Gebäudefassaden ist Thema von Kapitel 6. Zunächst wird die manuelle Texturierung beschrieben, um daraus ein automatisches Verfahren zu entwickeln. Nach der eher theoretischen Herangehensweise folgt die Beispielimplementierung eines konkreten Systems zur hardwarebasierten Texturextraktion. Mit diesem ist die Realisierung interaktiver Anwendungen möglich. Anschließend wird ein geometrischer Ansatz vorgestellt, welcher sich der schon in Kapitel 5 verwendeten Zellenzerlegung bedient.

Abschließend wird in Kapitel 7 eine Zusammenfassung der Arbeit gegeben und in einem Ausblick eine Weiterführung der Thematik diskutiert.

2 3D-Stadtmodelle

Neben dem digitalen Geländemodell als Referenzfläche ist die dreidimensionale Beschreibung von Gebäuden der wichtigste Bestandteil eines 3D-Stadtmodells. Zusätzlich können weitere Objektarten wie die Stadtmöblierung, Vegetation, Gewässer und Verkehrswege mitsamt Tunnel und Brücken im Modell mit enthalten sein.

Wie bereits in Kapitel 1 diskutiert, haben Anwendungen die unterschiedlichsten Anforderungen an die Detaillierung von 3D-Gebäudemodellen. Bevor in Kapitel 4 bis Kapitel 6 Verfahren zu deren automatischen Transformation in andere Detailstufen vorgestellt werden, soll zum besseren Verständnis zunächst auf die heutzutage vorherrschende Datengrundlage und deren Erfassung eingegangen werden. Beginnend mit der Festkörpermodellierung wird in Abschnitt 2.1 die Grundlagen zur geometrischen Modellierung allgemeiner 3D-Objekte behandelt. Es schließt sich ein Abriss über die gebräuchlichen Repräsentationsformen von 3D-Gebäuden an (Abschnitt 2.2). Die dabei gewählte Kategorisierung lässt sich sowohl durch die Modellierung als auch durch die existierenden Erfassungsmethoden motivieren. Dies ist darin begründet, dass sich die Verfahren zur Gebäudeextraktion der unterschiedlichen Modellierungsarten bedienen (siehe Abschnitt 2.3). Deshalb beeinflusst die jeweils genutzte Art der Modellierung unmittelbar die prinzipielle Vorgehensweise und damit die erreichbare Komplexität und Genauigkeit der resultierenden Entitäten. Da die flächendeckende Erfassung von Gebäuden in der Regel aus flugzeuggestützt erfassten Daten erfolgt, kann lediglich die Außenhülle ohne detaillierte Fassaden gewonnen werden. Hierfür sind terrestrische Bild- oder Laserdaten erforderlich. Wie man in Abschnitt 2.4 sehen wird, hat die Ausgestaltung von Fassaden aus terrestrischen Daten in den letzten Jahren große Aufmerksamkeit erfahren.

Immer mehr Städte und Kommunen lassen 3D-Stadtmodelle erfassen, auf die eine immer größere Zahl von Anwendungen zurückgreifen. Bisher verwalten die verschiedenen Erfassungswerkzeuge und Anwendungen ihre Daten jedoch häufig in proprietären Formaten. Ein einfacher Austausch von 3D-Stadtmodellen ist daher bislang nicht ohne weiteres möglich. In der jüngsten Vergangenheit wurde die Notwendigkeit nach einem normierten Austauschformat offenbar, die mit dem CityGML Standard erste Formen angenommen hat (Abschnitt 2.5).

2.1 Festkörpermodellierung

Die Festkörpermodellierung befasst sich mit der geometrischen Modellierung von dreidimensionalen Körpern, insbesondere von Menschenhand geschaffenen Objekten. Es werden nur geschlossene Repräsentationen von physikalisch festen Körpern erzeugt, die geeignet sind, geometrische Fragestellungen algorithmisch zu lösen.

Allgemeine Nachschlagewerke zum Thema Festkörpermodellierung sind beispielsweise [Hoffman 1989], [Foley et al. 1990] und [Samet 2006]. Hier sollen die wichtigsten Aspekte zusammenfassend dargestellt werden, wobei die Unterscheidung der Modellierungsarten von Mäntylä [1988] in Dekompositionsmodelle, Konstruktive Modelle und Randbeschreibung übernommen wurde.

2.1.1 Dekompositionsmodelle

Bei den Dekompositionsmodellen werden komplexe Körper durch die Kombination einfacher aneinander grenzender aber nicht überlappender Bausteine modelliert. Die einzelnen Varianten unterscheiden sich durch den Primitivtyp und die Art und Weise, wie deren Kombination gespeichert wird (vgl. Abbildung 2-1).

Die allgemeinste Form der Dekompositionsmodelle ist die **Zellenerlegung** (cell decomposition). Wie bei einem Kinderbaukasten erfolgt die Modellierung mittels elementarer als Zellen bezeichneter Primitive. Diese werden üblicherweise als parametrisierte Instanzen aus einer vordefinierten Menge von Zelltypen erzeugt. Sie dürfen beliebige auch gekrümmte Formen besitzen, müssen jedoch topologisch äquivalent einer Kugel

(d.h. ohne Löcher) sein. Ein komplexer Festkörper entsteht dann durch „Zusammenkleben“ von Zellen. Dabei handelt es sich um eine spezielle Vereinigungsoperation, bei der sich die Objekte in einem Eckpunkt oder an allen Punkten entlang einer Kante oder einer Fläche berühren müssen. Häufig wird diese Art der Repräsentation bei Analysen mittels der Methode der Finiten Elemente verwendet.

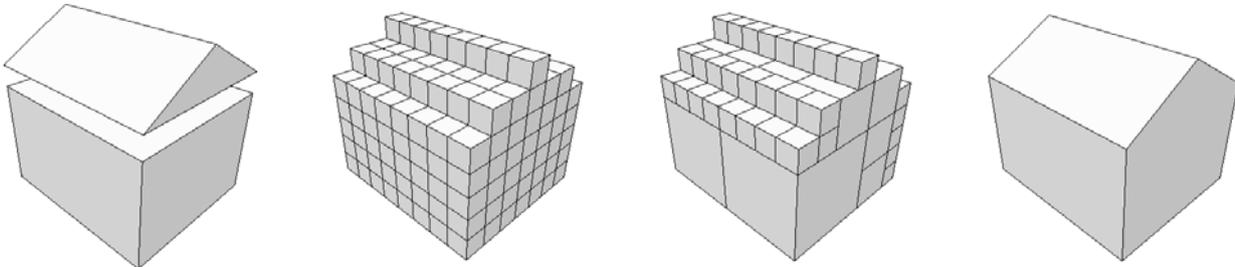


Abbildung 2-1: Von links nach rechts: 3D-Gebäudemodell in Zellenzerlegung, Enumerationsverfahren, Octree und BSP-Baum.

Eine spezielle Abwandlung der Zellenzerlegung ist das **Enumerationsverfahren** (spatial-occupancy enumeration). Hier wird der Festkörper mittels identischer Volumenelemente modelliert, welche in einem festen regulären Raster angeordnet sind. Diese Volumenelemente werden in Analogie zu Pixel (picture element) auch oft als Voxel (volume element) bezeichnet. Ein reguläres Raster kann zwar über diverse Grundformen erzeugt werden, die einfachste und am häufigsten verwendete ist der Würfel. Ein Objekt wird beim Enumerationsverfahren durch das Setzen der Volumenelemente, die sich innerhalb des Körpers befinden, repräsentiert. Da sich die Zugehörigkeit einer Zelle zum Objekt sehr leicht feststellen lässt, ist die Realisierung Boolescher Operatoren für diese Repräsentationsform äußerst einfach. Das Enumerationsverfahren wird sehr häufig für volumetrische Daten verwendet, z.B. in der Medizin im Bereich der Computertomographie. Hier ist nicht nur die Hülle von Interesse sondern auch das Innere des Objekts, welches sich sehr gut mittels Voxel darstellen lässt.

Durch die reguläre Rasterstruktur können Objekte allerdings nicht exakt sondern nur näherungsweise abgebildet werden. Dabei schränkt die Größe der Volumenelemente die räumliche Auflösung des Objekts ein. Da jeder Zelle individuelle Attributwerte wie beispielsweise eine Farbe zugewiesen werden kann, enthält ein Modell oftmals hochreduzante Information. Für eine feine Rastereinteilung ergeben sich so sehr schnell recht große Datenmengen. Eine Verringerung des Speicherbedarfs kann mittels hierarchischer Varianten wie dem **Octree** erreicht werden. Der Raum wird dabei sukzessiv in allen drei Dimensionen unterteilt, wodurch in jedem Schritt acht gleich große Kuben entstehen. Ein Kubus ist entweder vollständig innerhalb, vollständig außerhalb oder teilweise innerhalb des Objekts. Kuben, die sich nur teilweise innerhalb des Objekts befinden, werden bis zu einer maximalen Rekursionstiefe weiter unterteilt. Allgemeiner als der Octree ist der **Bintree**, der zwei statt acht Unterräume erzeugt. Die Unterteilung erfolgt wie beim Octree entlang der Koordinatenachsen, wodurch beide Verfahren zur exakten Modellierung beliebiger Objekte ebenfalls nicht besonders gut geeignet sind.

Die **BSP-Bäume** (binary space partitioning trees) lösen diese Restriktion und erlauben die rekursive Unterteilung des Raums durch Ebenen mit beliebiger Orientierung und Position. Das Modell wird als Graph dargestellt, in dessen Knoten die Schnittebene und in den jeweiligen Unterbäumen die beiden Teilräume vor und hinter der Ebene gespeichert werden. Konvexe Objekte müssen nicht unterteilt werden sondern können durch ein einziges Volumenelement modelliert sein. Jede planare Fläche definiert eine Schnittebene, so dass das Objekt vollständig hinter allen Ebenen liegt. Ursprünglich wurden BSP-Bäume in der Computergrafik genutzt um die Sichtbarkeit von Flächen in statischen Umgebungen zu bestimmen [Fuchs et al. 1980]. Thibault und Naylor [1987] haben später gezeigt, wie sich mit BSP-Bäumen beliebige Polyeder repräsentieren lassen. Das Objekt definiert sich durch die Kombination der homogenen Halbräume in den Blättern des Baums, die in Abhängigkeit vom Normalenvektor vollständig innerhalb oder außerhalb des Objekts liegen. Der Nachteil der Repräsentation ist die Unterteilung der Oberflächenpolygone, die sich durch die Erzeugung des Baums und die Anwendung Boolescher Mengenoperatoren ergibt. Eine vollständige Boolesche Algebra für 3D-BSP-Bäume wird beispielsweise von Naylor [1990] aufgestellt.

Dekompositionsmodelle sind wegen ihrer einfachen Struktur sehr gut für die Lösung numerischer Probleme geeignet. Allerdings ist deren Modellierung durch einen menschlichen Operateur kompliziert und nicht sehr intuitiv. Datensätze entstehen daher fast ausschließlich durch die Konvertierung aus anderen Repräsentationen. Geeigneter für die Modellierung sind die Konstruktiven Modelle.

2.1.2 Konstruktive Modelle

Die Konstruktiven Modelle bieten vielseitige Möglichkeiten um Grundelemente über Variantenbildung oder die Anwendung von Mengenoperatoren zu komplexen Objekten zu kombinieren (vgl. Abbildung 2-2).

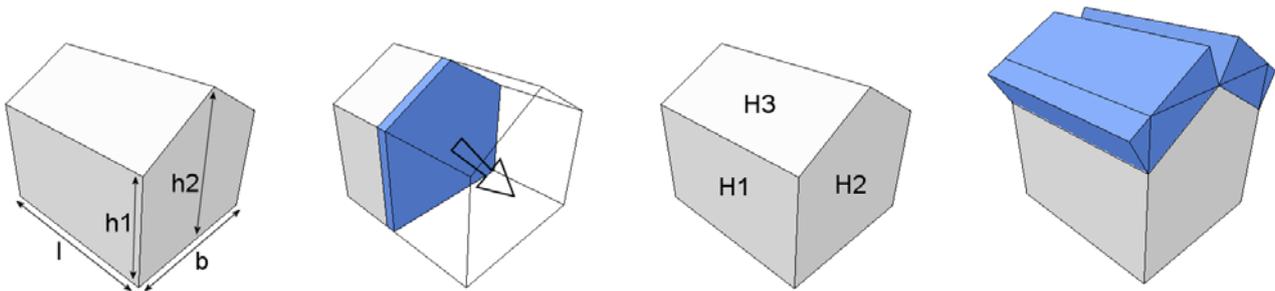


Abbildung 2-2: Von links nach rechts: 3D-Gebäudemodell in Parameterdarstellung, Sweep-Repräsentation, Modellierung mittels Halbräumen und Konstruktion mit Raumprimitiven.

Bei der **Parameterdarstellung** (primitive instancing) wird eine Menge anwendungsrelevanter 3D-Formen definiert, die durch eine feste Anzahl von Parametern charakterisiert sind. Für ein einfaches 3D-Gebäude mit Satteldach wären das beispielsweise die Länge und Breite des Grundrissrechtecks sowie die Trauf- und Giebelhöhe. Die Parameterdarstellung wird oft für relativ komplexe Objekte, z.B. für Zahnräder oder Bolzen, verwendet, die mittels Boolescher Operatoren nur mühsam als Kombination einfacher Elemente modellierbar sind, die sich jedoch mit nur wenigen Parametern beschreiben lassen. Bei der Parameterdarstellung ist die Kombination von Primitiven nicht vorgesehen, so dass die Modellierung damit auf die vordefinierten Objekttypen eingeschränkt ist.

Die **Sweep-Repräsentation** eines 3D-Objekts entsteht, indem eine planare Fläche im Raum bewegt wird. Das Volumen, das die Fläche über die Bewegung hinweg einnimmt, ist der Sweep-Körper. Die Bewegung kann eine Translation entlang einer Kurve oder die Rotation um eine Achse sein. Allgemeine Sweeps erlauben auch die Kombination der beiden Bewegungsarten sowie eine Änderung der formgebenden Fläche. Die Repräsentation mittels Sweeps eignet sich gut für symmetrische Objekte. Allgemeine Körper können mit dieser Technik jedoch nicht erzeugt werden, da reguläre boolesche Operationen nicht auf Sweeps angewendet werden können ohne das Objekt vorher in eine andere Repräsentation umzuwandeln.

Eine weitere Methode ist die **Modellierung mittels Halbräumen** (Half Spaces). Ein Halbraum ist eine Funktion $f(X) = 0$, welche die Punkte $X=(x, y, z)$ des dreidimensionalen Raums in die zwei Untermengen $f(X) \geq 0$ und $f(X) \leq 0$ aufteilt. Beispielsweise definieren die beiden Funktionen

$$ax + by + cz + d \geq 0$$

$$x^2 + y^2 - r^2 \leq 0$$

den planaren Halbraum mit allen Punkte auf der positiven Seite der Ebene $ax+by+cz+d=0$ und den zylindrischen Halbraum mit allen Punkte innerhalb des unendlichen Zylinders entlang der z-Achse mit Radius r (die Begrenzungsfläche jeweils mit eingeschlossen). Andere Halbräume von Interesse sind Kugel, Kegel, Torus, etc. Da Halbräume Punktemengen sind, können auf sie die Booleschen Operatoren Vereinigung (\cup), Verschneidung (\cap) und Differenz (\setminus) angewendet werden. Demzufolge werden Objekte durch die Kombination von Halbrauminstanzen mittels Boolescher Operatoren kombiniert. Zum Beispiel kann das Gebäude aus Abbildung 2-2 durch die Verschneidung (\cap) der sieben planaren Halbräumen H1 bis H7 beschrieben werden.

Die Modellierung mittels Halbräumen ist wegen der häufig unbegrenzten Grundprimitive für den Menschen nicht sehr intuitiv. Eine Methode zur Modellierung mit begrenzten Körpern ist die **Konstruktion mit Raum-**

primitiven (constructive solid geometry). Ein Objekt wird durch die Boolesche Kombination meist parametrisierter Grundelemente erzeugt. Die Konstruktion des Objekts wird in einem Baum gespeichert. Dabei sind die Grundprimitive die Blätter des Baums, die durch die Operatoren hierarchisch verknüpft sind. Zu den Operatoren gehören nicht nur Booleschen Operatoren sondern auch Transformationen wie Translation, Rotation und Skalierung. Sofern ausschließlich reguläre Mengenoperatoren Verwendung finden, entstehen bei der Modellierung mit Raumprimitiven nur gültige Körper. Ist die Endlichkeit Voraussetzung für einen gültigen Körper, kann dies bei der Halbraummodellierung nicht gewährleistet werden, da Halbräume für sich genommen schon keine gültigen Körper darstellen. Da die Zellenzerlegung ein Sonderfall der Konstruktion mit Raumprimitiven darstellt, zeigt die Abbildung 2-2 auch ein gültiges Beispiel für diese Modellierungsart. Sind nur Quader als Grundelemente verfügbar, so kann das Gebäudemodell auch als die Differenz eines großen Quaders und zwei kleinen gedrehten Quadern modelliert werden (siehe Abbildung 2-2).

Komplexe Festkörper lassen sich mithilfe der konstruktiven Modelle mit einfachen Mitteln aufbauen und äußerst präzise beschreiben. Für deren graphische Ausgabe empfiehlt es sich jedoch die Daten zunächst in eine polygonale Repräsentationsform wie der Randbeschreibung zu konvertieren. Andere Repräsentationsformen sind mit modernen Echtzeitrenderern häufig nicht direkt darstellbar.

2.1.3 Randbeschreibung

Die Definition eines Körpers erfolgt bei der **Randbeschreibung** (boundary representation) durch die umhüllende Begrenzungsfläche (siehe Abbildung 2-3). Diese setzen sich durch die topologischen Primitive Knoten, Kanten und Flächen hierarchisch zusammen. Im Fall der Polyedermodelle bestehen die Flächen aus planaren Polygonen, welche durch eine Menge von Knoten umgeben sind. Deren Reihenfolge bestimmt die Außenseite des Körpers. Die einfachste Form eine Fläche in Randbeschreibung zu strukturieren, ist die Beschreibung durch eine Sequenz dreidimensionaler Koordination. Aufwendigere Datenstrukturen wie Winged-Edge [Baumgart 1974], Half-Edge [Weiler 1985], Quad-Edge [Guibas und Stolffy 1985], etc. speichern zusätzlich auch explizit die topologischen Relationen, so dass ein effizienter Zugriff auf Nachbarflächen, vorherige und nachfolgende Kanten, usw. möglich ist.

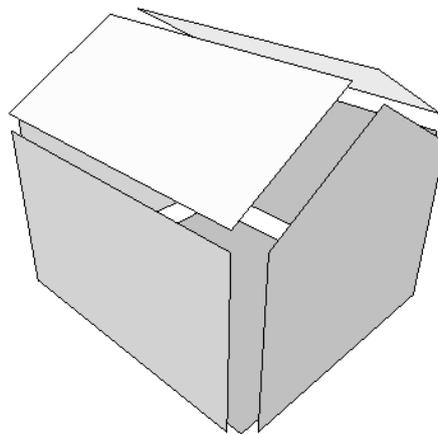


Abbildung 2-3: Randbeschreibung eines 3D-Gebäudemodells mit Satteldach.

Gewöhnlich besitzen 3D-Gebäudemodelle keine gekrümmte Oberfläche, d.h. sie gehören zur speziellen Gruppe der Polyeder. Deren Randbeschreibung ist eine eindeutige Zerlegung in maximal zusammenhängende Teilflächen. Sämtliche Flächen werden explizit beschrieben, was die Analyse sowohl von Einzelobjekten als auch von Objektgruppen vereinfacht. Die Konvertierung aus einer anderen Repräsentationsform ist algorithmisch leicht möglich, wenngleich eine stabile Implementierung aufgrund numerischer Ungenauigkeiten nicht trivial erscheint (vgl. [Hoffman 1989]). Auch wenn tiefer gehende Analysen und Abfragen noch nicht möglich sind, können Datenbanken und Geo-Informationssysteme mittlerweile raumbezogene 3D-Daten in Randbeschreibung zumindest verwalten. Zu deren Austausch und Speicherung spezifiziert das OpenGIS Konsortium (OGC) die Geography Markup Language (GML) [OpenGIS 2004].

2.2 3D-Gebäudemodelle

Die Modellierung von 3D-Gebäudemodellen kann in allen vorgestellten Varianten der Festkörpermodellierung erfolgen. Aufgrund des konstruktiven Charakters wird häufig den modellgetriebenen (top down) Techniken, wie beispielsweise der Parameterdarstellung und der Konstruktion mittels Raumprimitiven, der Vorzug gegeben. Sie sind intuitiv verständlich und führen zu relativ regelmäßigen und einfachen Modellen. Mit der notwendigen Unterstützung durch entsprechende Softwarewerkzeuge kann aber auch die datengetriebene (bottom up) Modellierung in Randbeschreibung erfolgen. Diese ist weniger restriktiv und ermöglicht eine bessere Annäherung an die reale Gebäudeform, führt jedoch auch zu komplexeren und unregelmäßigeren Geometrien. Wie Steinle [2005] anmerkt, sind diese Eigenschaften im Wesentlichen auch im gewählten Abstraktionsgrad begründet, da durch eine grobe Erfassung eher einfachere und symmetrische Modelle entstehen. Unabhängig von der eigentlichen Modellierung werden die resultierenden Gebäudeobjekte anschließend zumeist im Rahmen der Datenspeicherung und des Datenaustauschs in eine Randbeschreibung konvertiert. Unabhängig von der Herkunft und der Modellierungsart, die bei der Erfassung verwendet wurde, klassifiziert Förstner [1999] wie folgt Gebäudemodelle aufgrund ihrer Gebäudeform und der damit verbundenen Komplexität (vgl. Abbildung 2-4):

Parametrische Gebäudemodelle bestehen aus meist rechteckigen Grundrissen mit aufgesetzten Standarddachformen wie beispielsweise Flach-, Pult-, Sattel-, Mansard-, Walmdach, etc. Deren Variationen sind durch die Anzahl der Parameter beschränkt, so dass sich nur einzelstehende Gebäude modellieren lassen. Dafür sind die geometrischen und topologischen Eigenschaften der Modelle implizit durch die Vorlagen gegeben. Diese beinhalten je nach Modell z.B. identische Traufhöhen und symmetrische Dachneigungen.

Kombinierte parametrische Gebäudemodelle entstehen durch die Vereinigung der einfachen parametrischen Primitive zu komplexen Modellen. Sofern auch das bei Einzelgebäuden selten vorkommende Zeltdach als Primitiv gegeben ist, sind die meisten Gebäudeformen inklusive einfacher Türme modellierbar. Im Gesamtmodell dürfen durchaus unterschiedliche Trauf- und Giebelhöhen existieren, für die Teilmodelle gelten implizit jedoch wieder dieselben geometrischen und topologischen Eigenschaften wie oben. Durch die eingeschränkte Auswahl an Primitiven ist die Modellierung von Gebäuden mit runden und spitzwinkligen Grundrissen sowie ungewöhnliche Dachformen meist nicht möglich.

Prismatische Gebäudemodelle, oder auch Blockmodelle genannt, entstehen durch einfache Extrusion. Sie besitzen beliebig komplexe Grundrisse und ein Flachdach konstanter Höhe. Wegen der einfachen Modellierung aus 2D-Grundrissdaten ist dieser Typ weit verbreitet. Es lassen sich aber auf diese Weise nur wenige Gebäude exakt modellieren.

Polyedermodelle bestehen aus einer Menge planarer Polygone. Durch diese lässt sich die überwiegende Mehrzahl der Gebäude mitsamt Erkern, Gauben, Schornsteine, etc. beschreiben. Da die Flächen beliebig angeordnet sein dürfen, müssen geometrische Bedingungen (einheitliche Höhen, Symmetrien, Parallelität, Rechtwinkligkeit, usw.) explizit angegeben bzw. modelliert werden.

Die Gebäudemodelle dieser vier Klassen bestehen aus planaren Flächen. Für Gebäude mit gekrümmten Flächen ergänzt Brenner [2000] die Auflistung um zwei weitere Gebäudeklassen:

Regelgeometrische Flächen vervollständigt die (kombinierten) parametrischen Gebäudemodelle um Primitive mit gekrümmten Dachstrukturen (Tonnen-, Kuppel-, Kegeldach, etc.). Gebäude mit solchen Formen werden bislang jedoch nur manuell erfasst.

Freiformflächen können beliebige gekrümmte Formen besitzen. Da nur wenige meist prominente Gebäude wie Stadien oder Konzertsäle durch die anderen Modelle nicht beschreibbar sind, wurden solche Gebäudemodelle für die Rekonstruktion noch nicht in Erwägung gezogen.

Nachdem die Modellierung allgemeiner Körper erörtert und eine Klassifikation von Gebäudemodellen anhand der Komplexität durchgeführt wurde, soll nun der Frage nachgegangen werden, wie solche Modelle möglichst automatisch aus Sensordaten extrahiert werden können.

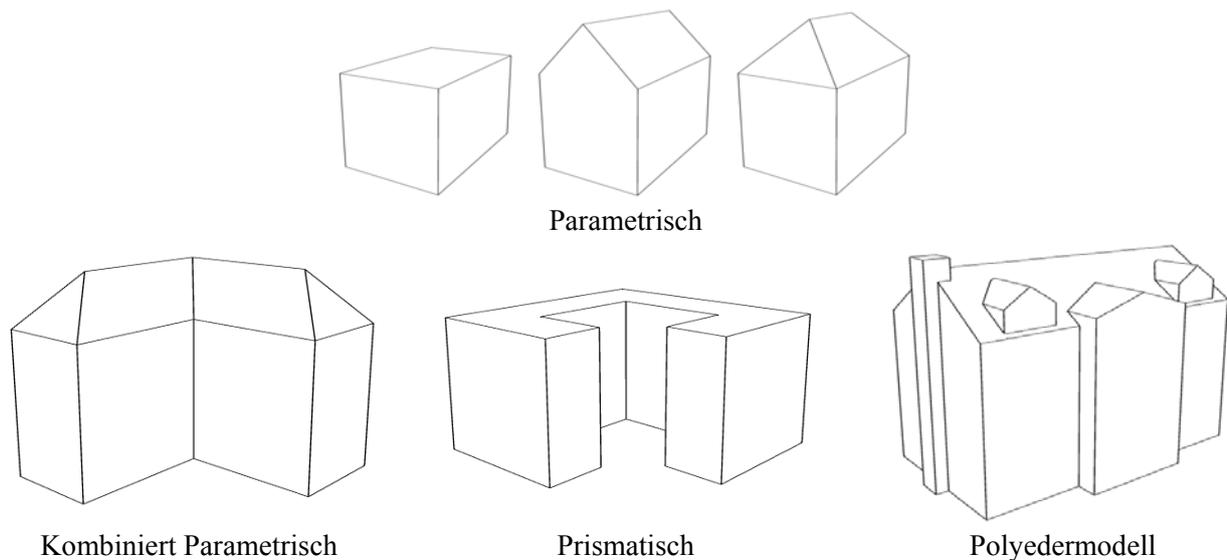


Abbildung 2-4: Klassifikation von 3D-Gebäudemodellen nach Förstner [1999] und Brenner [2000].

2.3 Gebäudeextraktion

Eine großflächige Erfassung von 3D-Gebäudemodellen erfolgt über die Messung und Interpretation von aus der Luft erfasster Bild- und Sensordaten. Wie zahlreiche aktuelle Publikationen belegen, ist dieser als Gebäudeextraktion bezeichnete Prozess ein noch immer gegenwärtiges Forschungsthema in der Photogrammetrie und Fernerkundung. Deren detaillierte und umfassende Beschreibung ist aufgrund der Fülle der mittlerweile bekannten Verfahren und Algorithmen im Rahmen dieser Arbeit nur schwer möglich. Dennoch sollen die Konzepte kurz erläutert werden, da die aus der Gebäudeextraktion resultierenden 3D-Gebäudemodelle die Datengrundlage für die in dieser Arbeit vorgeschlagenen Verfahren darstellen. Für eine tiefer gehende Auseinandersetzung mit dem Thema sei beispielhaft auf die Arbeiten von Brenner [2000],[2005], Rottensteiner [2001] und Hofmann [2005] verwiesen.

Die Gebäudeextraktion unterteilt sich gemäß Brunn [1998] in die Gebäudedetektion und die Gebäuderekonstruktion. In der Gebäudedetektion werden die relevanten Regionen in den Eingabedaten ermittelt, d.h. es wird in den vorhandenen Datenmaterialien nach Gebäudegrenzen gesucht. Dies kann beispielsweise in Luftbildern [Baltsavias und Mason 1997], [Ameri 2000], Satellitenbildern [Sohn und Dowman 2001], Höhendaten (Laserdaten, Stereobildauswertung, etc.) [Baltsavias et al. 1995] [Henricsson et al. 1996], 2D-GIS-Daten [Haala et al. 1998], SAR [Sörgel 2003] (siehe Abbildung 2-5) erfolgen oder durch interaktive Bestimmung der Gebäuderegion. In der sich anschließenden Gebäuderekonstruktion werden die geometrischen Parameter der Gebäude ermittelt um daraus 3D-Festkörper zu generieren.



Abbildung 2-5: Luftbild und SAR Bild mit überlagerten Grundriss (aus [Stilla et al. 2002]).

Verfahren zur Rekonstruktion von 3D-Gebäudemodellen lassen sich nach Brenner [2000] aufgrund folgender Kriterien einteilen:

- **Verwendete Datenquellen:** Hier kommen im Prinzip dieselben Datenquellen aus der Fernerkundung zum Einsatz wie auch schon bei der Gebäudedetektion: Luft-, Satelliten-, Farb- und Multispektralbilder, Laserscanning- und Radardaten. Hinzu kommen vorverarbeitete Daten aus der Photogrammetrie, klassifizierte Daten und 2D-GIS-Daten.
- **Gebäudemodellierung:** Die Einteilung erfolgt aufgrund der eingesetzten Festkörpermodellierung (siehe Abschnitt 2.1).
- **Funktionsweise:** semiautomatisch oder automatisch. Die Forschung hat sich historisch zunächst auf Verfahren zur automatischen Rekonstruktion aus Luftbildern konzentriert. Wegen der Komplexität der Aufgabe gelang jedoch keine ausreichende Genauigkeit und Stabilität. Die Entwicklung und Erfassung hat sich daher schwerpunktmäßig hin zu den semiautomatischen Verfahren verlagert. Erst durch die Verfügbarkeit von luftgestütztem Laserscanning kam in den letzten Jahren wieder vermehrt Interesse an automatischen Verfahren auf.

Im Folgenden soll exemplarisch näher auf einige Rekonstruktionsverfahren eingegangen werden. Die Aufzählung erhebt keinen Anspruch auf Vollständigkeit, sondern soll vielmehr ein möglichst breites Spektrum an unterschiedlichen Konzepten und Algorithmen abdecken.

Die Dissertation von Haala [1996] beschreibt ein automatisches Verfahren zur Rekonstruktion von Gebäuden aus Stereobildpaaren. Mittels automatischer Stereobildzuordnung wird aus den Bildpaaren ein Digitales Höhenmodell als zusätzliche Datenquelle erzeugt. Alternativ kann auch ein bildunabhängiges DHM aus LIDAR-Daten verwendet werden (siehe [Haala 1997]). Zur Rekonstruktion extrahiert das Verfahren zunächst Objektgrenzen und Bruchkanten der Oberflächen aus den Bildern und bestimmt daraus durch Stereozuordnung dreidimensionale Liniensegmente. Sind die Höhendaten zuverlässig und dicht genug erfasst, so kann die Extraktion der Bruchkanten auch über eine Krümmungsanalyse der Höhendaten erfolgen. Die gefundenen Segmente werden anschließend durch eine Analyse der Höhendaten gefiltert. Dabei wird angenommen, dass Gebäude eine begrenzte räumliche Ausdehnung haben und aus der Geländeoberfläche herausragen. Sie bilden im DHM also ein lokal begrenztes Höhenmaximum, welches über die Verwendung von Höhenlinien oder einer morphologischen Bearbeitung des DHM gefunden werden kann. Durch Gruppierung der extrahierten linearen Segmente erfolgt zusätzlich eine Suche nach gebäudetypischen Strukturen. Dadurch werden im anschließenden Zuordnungsprozess besonders solche Kombinationen linearer Segmente beachtet, denen mit großer Wahrscheinlichkeit ein Gebäude zugrunde liegt. Zur Rekonstruktion des Gebäudes werden mögliche Korrespondenzen zwischen den extrahierten Segmenten und den Linien der Objektmodelle definiert. Das Verfahren verwendet für die Modellierung der Gebäude eine parametrisierte Darstellung und unterscheidet zwischen den Dachformen Walm-, Sattel- und Pultdach. Durch eine Ausgleichung nach kleinsten Quadraten werden die Formparameter (Länge, Breite, Höhe und Neigungswinkel) so bestimmt, dass die Differenzen zwischen den Modelllinien und den zugeordneten Liniensegmenten minimal werden. Die endgültige Auswahl des Rekonstruktionsergebnisses geschieht über eine Verifikation der Zuordnungshypothese mit Hilfe des resultierenden Fehlers aus der Parameterschätzung. Henricsson [1996] zeigt einen ähnlichen Ansatz, bei dem zusätzlich die Farbinformationen entlang der extrahierten Konturen ausgewertet werden.

Da sich die Rekonstruktion von Gebäuden ausschließlich aus Luftbildern als schwierig erwiesen hat, wurden für die Interpretation vermehrt zusätzliche Datenquellen mit einbezogen. Suveg und Vosselman [2004] verwenden GIS-Daten in Form von 2D-Gebäudegrundrissen um mögliche Lösungen einzuschränken. Ähnlich dem Ansatz von Haala wird eine Bibliothek von einfachen Gebäudeformen wie Walm-, Sattel- und Pultdach verwendet, um über Hypothesenbildung und -verifikation die am besten passende Gebäudeform zu finden. Damit sich auch komplexe Gebäudeformen rekonstruieren lassen, können die einfachen Gebäudeformen auch zu komplexen Gebilden kombiniert werden. Im ersten Schritt wird daher der Gebäudegrundriss in eine disjunkte Menge von Rechtecken zerlegt. Üblicherweise existieren für diese Zerlegung mehrere Möglichkeiten. Für jede Zerlegung werden daher Hypothesen über die dreidimensionale Dachform der Rechtecke aufgestellt und mittels Rückprojektion auf das Luftbild ausgewertet. Das Gesamtgebäude aus einer Zerlegung definiert sich nun als CSG Repräsentation der Rechtecke mit der besten Passform. In einem weiteren Verifikationsschritt wird das Gesamtgebäude an die Bilddaten mittels einer Ausgleichung nach kleinsten Quadraten angepasst. Um das Ergebnis zu verbessern, werden geometrische Wechselbeziehungen zwischen den Gebäudeprimitiven zusätzlich als Zwangsbedingungen mit in den Prozess integriert. Die resultierenden 3D-Primitive ergeben zusammengenommen das Gebäudemodell in einer Repräsentation als Zellenzerlegung.

Der CyberCity Modeler ist ein semiautomatisches Werkzeug, um die Topologie von 3D-Gebäudeobjekten zu erzeugen [Wang und Grün 1998]. Das Rekonstruktionsverfahren basiert auf der manuellen Stereomessung einer 3D-Punktwolke aus Satelliten- und Luftbildern. Während der Messung werden die zu einem Gebäudeobjekt zugehörigen Punkte in Randpunkte und innere Punkte unterschieden und entsprechend gekennzeichnet (siehe Abbildung 2-6). Vor der automatischen Auswertung müssen die Randpunkte in eine vorgegebenen Reihenfolge gebracht werden. Danach wird die Topologie des Dachs über einen Relaxationsprozess bestimmt. Mittels einer Ausgleichung nach kleinsten Quadraten werden die entstandenen Flächen an die gemessenen Punkte angepasst. Dabei werden Beobachtungsgleichungen in den Ausgleichungsprozess eingefügt um die Orthogonalität von Linienpaaren zu berichtigen. Das Ergebnis ist ein 3D-Gebäudemodell in Randbeschreibung. Da die Modellierung über Punkte und planaren Flächen erfolgt, können beliebige polygonale Dachstrukturen erzeugt werden.

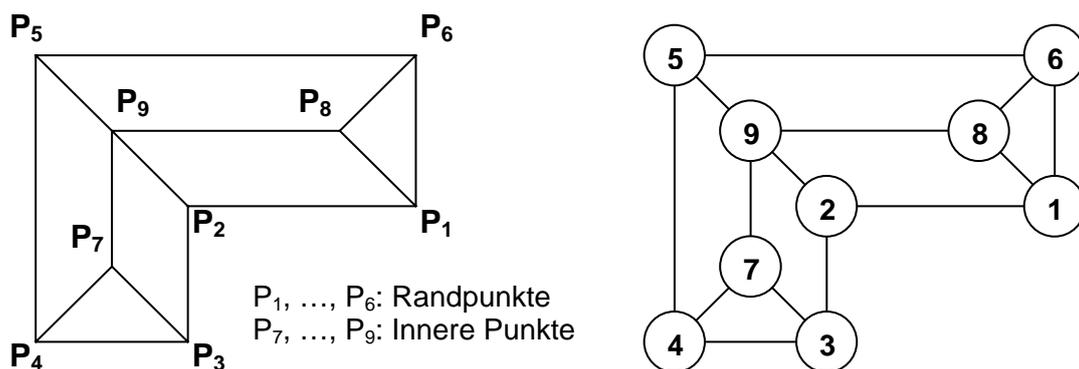


Abbildung 2-6: Punktdefinition in CyberCity Modeler und zugehöriger Graph.

Ein weiterer semiautomatischer Ansatz wird von Wolf [1999] beschrieben. Zunächst wird mittels manueller Stereoauswertung aus Farbluftbildern ein reguläres Höhenraster erzeugt sowie Bruchkanten (Abhänge, Wände, Rampen) und Höhenpunkte (Scheitelpunkte, Senken) gemessen. Aus diesen Daten wird ein DGM berechnet und mit den Gebäudegrundrissdaten aus der automatisierten Liegenschaftskarte (ALK) verschnitten. Das Ergebnis ist ein Ring von Fassadenpolygonen. Für die Rekonstruktion der Dachstruktur wird aus einer Bibliothek eine Basisform wie Flach-, Giebel-, Walm-, Pyramiden-, Tonnen-, Shed-, Kuppeldach, etc. ausgewählt und die entsprechenden Dachparameter im Luftbild gemessen. Anschließend wird die Dachform mit dem Fassadenring verschnitten um eine 3D-Randbeschreibung zu erhalten. Die Dachform von komplexen Gebäuden wie z.B. Schlösser, Türme oder Kirchen werden aus den Basisformen zusammengesetzt (siehe Abbildung 2-7).

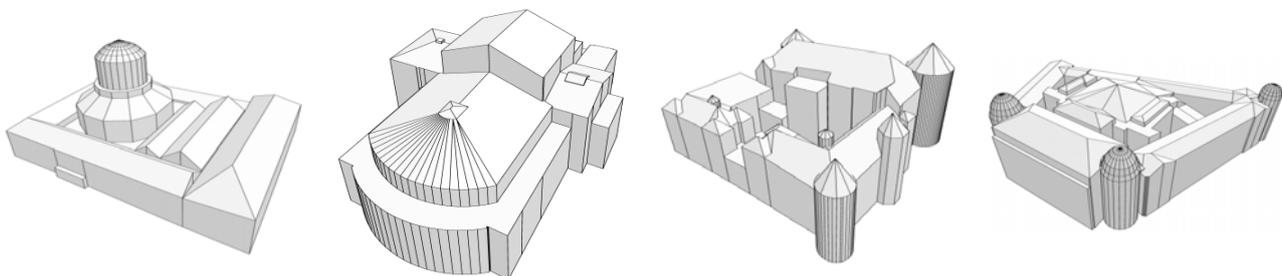


Abbildung 2-7: Ausgewählte Gebäudemodelle aus dem Stuttgarter 3D-Stadtmodell als regelgeometrische Flächen.

Das kommerzielle Softwaresystem inJECT basiert auf den Ansätzen von Gülch et al. [1998], [1999]. Gebäude werden mit Hilfe einer festgelegten Zahl von parametrischen Gebäudeprimitiven wie Flach-, Walm-, Sattel-, Pultdach, etc. modelliert. Die Auswahl erfolgt manuell durch den Operator, woraufhin das gewählte Gebäudeprimitiv in einem oder mehreren Bildern als Drahtgittermodell überblendet wird. Interaktiv können nun die Parameter des Gebäudes angepasst werden. Der Operator wird dabei von der Software durch den Messprozess geführt. Für ein Satteldach müssen beispielsweise nur zwei Giebelpunkte in einem Bild gemessen

sen werden. Das System versucht daraufhin automatisch die korrespondierenden Punkte in den anderen Bildern zu finden und die Parameter zu berechnen. Anschließend bekommt der Operateur das Ergebnis zur Bewertung und manuellen Korrektur angezeigt. Komplexe Gebäude müssen aufgeteilt und unabhängig voneinander modelliert werden. Das endgültige Gebäudemodell entsteht in einem CSG-Prozess durch Vereinigung der Teile.

Brenner und Haala [1998] beschreiben einen automatischen Ansatz zur Rekonstruktion von Gebäuden aus 2D-Grundrissen und LIDAR-Daten (vgl. Abbildung 2-8). Der Grundriss wird mittels eines heuristischen Algorithmus in rechteckige Primitive unterteilt. Im Gegensatz zu Suveg und Vosselman [2004] ist die Menge der Primitive jedoch nicht disjunkt, d.h. die Rechtecke überlappen sich. Aus einer festen Menge von Standardtypen werden für jedes Rechteck 3D-Primitive in parametrisierter Darstellung instantiiert. Die Parameter der Primitive werden mit Hilfe eines DOM in Rasterform geschätzt, welches durch Interpolation der LIDAR-Daten erzeugt wurde. Der Primitivtyp mit der besten Passform wird anschließend selektiert. In einer sich anschließenden semiautomatischen Erweiterung kann die Auswahl des Primitivtyps und die Parameterwerte noch verändert werden, bevor das endgültige Gebäudemodell durch die Boolesche Vereinigung der 3D-Primitive gewonnen wird.

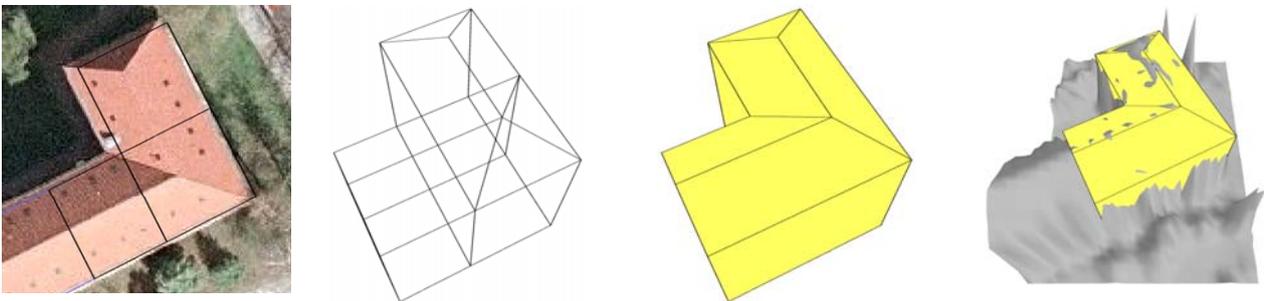


Abbildung 2-8: Orthophoto mit überlagerter Rechteckszerlegung des Grundrisses, geschätzte Primitive, Vereinigung zu fertigem 3D-Gebäudemodell und Überlagerung mit DOM.

Problematisch bei dem Ansatz ist vor allem die Rechteckszerlegung des Grundrisses. Während stumpfe Winkel sich noch durch geschicktes Drehen der Rechtecke realisieren lassen, ist die Modellierung von spitzen Winkeln schlichtweg unmöglich. Selbst wenn eine gute Approximation des Grundrisses gefunden wurde, lässt sich aus der Menge der zur Verfügung stehenden Standardprimitiven nicht immer eine geschlossene Dachfläche bilden (siehe Abbildung 2-9). Außerdem müssen komplexe Grundrisspolygone durch sehr viele Rechtecke angenähert werden, was zu einer großen Menge von sich überschneidenden 3D-Primitiven führt (siehe Abbildung 2-9). Aufgrund kleiner Ungenauigkeiten in der Lage der Rechtecke kann es vor allem bei komplexen Dachstrukturen zu Schwierigkeiten bei der Booleschen Vereinigung der Primitive kommen. Um dem entgegenzutreten, schlägt Brenner [2004] vor, die Rechtecke mittels Zwangsbedingungen miteinander zu verknüpfen.

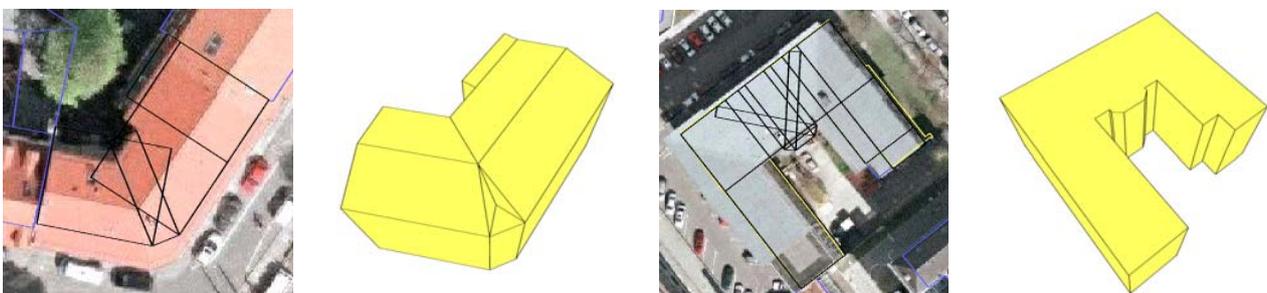


Abbildung 2-9: Orthophotos mit überlagerten Rechteckszerlegungen der Grundrisse und daraus resultierende 3D-Gebäudemodelle.

Die automatische Methode von Hofmann [2005] verwendet 3D-Punktwolken aus LIDAR-Daten, die mittels einer Segmentierung jeweils für ein Gebäude vorselektiert und extrahiert werden. Aus diesen Höhendaten

wird zunächst eine 2.5D-Dreiecksvermaschung nach Delaunay gerechnet. Anschließend werden für jedes Dreieck die Lageparameter im Raum (Ausrichtung, Neigungsgrad und senkrechter Abstand zum Schwerpunkt der Punktwolke) berechnet und in einem Parameterraum aufgetragen. Über eine Clusteranalyse können nun die Ebenen der Dachstruktur bestimmt werden. Um die Topologie zwischen den Dachebenen aufzubauen, werden Gauben detektiert sowie die Ebenen von First- und Querelementen gruppiert. Anschließend ist die Verschneidung der Ebenen möglich, um die polygonalen Dachflächen zu erhalten. Der fertigen Dachstruktur werden abschließend noch Fassadenwände zugefügt, damit ein komplettes 3D-Gebäudemodell in Randbeschreibung entsteht.

Ein konzeptionell ähnlicher Ansatz wird von Ma [2004] vorgestellt. Nach der Detektion der Gebäude mittels Segmentierung der LIDAR-Daten werden Ebenen mit Hilfe einer Clusteranalyse bestimmt und aufgrund erkannter Nachbarschaftsbeziehungen zu planaren Flächen verschnitten. Mit Hilfe einer automatischen Stereoauswertung von Luftbildern wird die horizontale Genauigkeit der Gebäudekanten abschließend verbessert.

Sörgel [2003] beschreibt ein Verfahren basierend auf SAR-Daten. Mit Hilfe von Verfahren aus der Bildverarbeitung werden Primitivobjekte wie z.B. Linien und Regionen in den Betragsbildern segmentiert. Damit dies überhaupt möglich ist, müssen die Daten jedoch vorher mittels Speckle-Filter und einer Kappung von Signalspitzen aufbereitet werden. Anschließend werden iterativ Hinweise auf Gebäude und Straßen erzeugt. Die Rekonstruktion des Grundrisses erfolgt dabei als Rechteck, als rechtwinklige Struktur oder als Polygon beliebiger Struktur. Des Weiteren werden über eine Analyse der InSAR-Höhendaten und der Schatten Hypothesen über die Dachstruktur aufgestellt. Unterstützt werden die Dachtypen Flach-, Pult- und Giebeldach, sowie prismatische Gebäude. Dabei handelt es sich bei den ersten drei Dachformen um parametrische Gebäudeprimitive, deren Parameter aus den SAR-Daten gewonnen werden. Prismatische Gebäude sind Flachdachkomplexe, die bei jeder Iteration wachsen können. Beim Übergang vom SAR-Koordinatensystem in das 3D-Weltkoordinatensystem werden die Gebäudehinweise in eine 3D-Szenenbeschreibung transformiert. Diese dient als Grundlage für eine Simulation mit den Parametern der realen SAR-Daten. Durch Vergleich der Ergebnisse aus der Simulation mit den SAR-Daten werden als falsch erkannte Hinweise verworfen und neue erzeugt. Abweichungen zwischen den Ergebnissen und den Daten werden dabei sukzessiv beseitigt. Bei der Auswertung von Multiaspekt Daten werden die einzelnen Rekonstruktionsergebnisse fusioniert. Dadurch lassen sich verdeckte Bereiche ergänzen und Ausreißer eliminieren. Durch aktuelle Entwicklungen im Bereich der SAR-Sensoren sind mittlerweile Bilddaten mit deutlich höheren Auflösungen bis zu einer Größenordnung von einem Dezimeter verfügbar. Die Möglichkeiten, die sich damit im Bereich der Gebäuderekonstruktion ergeben, werden von Michaelsen et al. [2005] diskutiert.

Die Genauigkeitsuntersuchung von Kaartinen et al. [2005] vergleicht zwölf aktuelle Gebäuderekonstruktionsverfahren. Es wurden sowohl automatische als auch semiautomatische Ansätze betrachtet, die auf der photogrammetrischen Stereoauswertung, dem Laserscanning und hybriden Methoden basieren. Es wurden gescannte analoge Luftbilder mit einer Bodenpixelauflösung zwischen 5,6 cm und 7,4 cm zur Verfügung gestellt. Die Untersuchung kommt zu dem Ergebnis, dass durch photogrammetrische Auswertung eine Genauigkeit der resultierenden Gebäudemodelle von ca. 20 cm in der Lage erreicht werden kann. Dies ist natürlich abhängig von der Qualität und Auflösung der Bilddaten. Zum Vergleich wird der Rekonstruktion von Gebäudeumrissen per Laserscanning nur eine Genauigkeit von ca. 60 cm bestätigt. Die Hauptgründe für das schlechtere Ergebnis scheinen in der Punktdichte der Daten, den Abschattungen durch Bäume und in der Komplexität der Strukturen begründet zu sein. Bei der Bestimmung von Höhen wurden jedoch Genauigkeiten im Bereich der Punktdichte erreicht. Für Laserscanning-Daten mit hoher Dichte (10-20 Punkte pro m²) wurde mit allen untersuchten Verfahren eine Genauigkeit besser als 20 cm erzielt. Auch die Dachneigungen konnten aus Laserscanning-Daten genauer erfasst werden, wobei eine starke Abhängigkeit vom Automationsgrad und der eingesetzten Methode beobachtet werden konnte. Für das für die Arbeit zur Verfügung stehende 3D-Stadtmodell von Stuttgart wird von Wolf [1999] eine Genauigkeit der gemessenen Punkte von ca. ± 8 -10 cm in der Lage und ca. ± 15 -20 cm in der Höhe angegeben.

2.4 Fassadenrekonstruktion

Neben Verfahren, die auf flugzeuggestützt erfasste Daten basieren, werden auch Ansätze zur Gebäuderekonstruktion aus terrestrischen Daten entwickelt. Gebäude werden mitsamt Fassade modelliert, wobei der Schwerpunkt zunächst in der bildbasierten Rekonstruktion lag. In einem frühen semiautomatischen Ansatz

von Debevec et al. [1996] werden Gebäude mit Fassadenstrukturen aus parametrisierten Grundprimitiven zusammengesetzt. Eine Parameterausgleichung passt die Primitive mittels Stereo-Matching an interaktiv gekennzeichnete Kanten in Bildfolgen ein. Nicht vollständig sichtbare Primitive werden von Dorffner und Lorkert [1998] durch die Integration von Hypothesen in eine hybride Bündelblockausgleichung generiert. Einen ähnlichen Ansatz zeigt El-Hakim [2002] im Anwendungsfeld der Dokumentation von Kulturdenkmälern und Landmarken. Regelmäßige Formen wie Blöcke, Bögen, Säulen und Fenster werden interaktiv aus gemessenen Startpunkten (seed points) modelliert. Fehlende Punkte werden dabei automatisch dem Modell hinzugefügt. Lee und Nevatia [2003] strukturieren ihre Architekturmodelle hierarchisch in Grundform und detaillierte Geometrien. Um Objekte vollautomatisch zu rekonstruieren, werden häufig Zwangsbedingungen in die Auswertung integriert, die sich aus Mutmaßungen über Objektform und Flächenverhältnisse oder aus Sichtbarkeitsanalysen ergeben [Grossmann und Santos-Viktor 2005], [Dick et al. 2004], [Werner und Zisserman 2002], [Hilton 2005]. Coorg und Teller [1999] sowie Wang et al. [2002] beschränken sich auf vertikale Fassaden und erfassen texturierte Blockgebäude. Die vollständige Automation bildbasierter Modellierung ohne jedwede Benutzereingabe ist jedoch bisher nicht vollständig gelöst [El-Hakim 2006].

Demgegenüber stehen die automatischen Verfahren zur Erfassung von zunächst allgemeinen Oberflächen. Gute Ergebnisse konnten beispielsweise mittels Space Carving [Kutulakos und Seitz 2000], Voxel Coloring [Seitz und Dyer 1999] oder Level Set Methoden [Faugeras und Keriven 1998] erreicht werden. Bildbasierte Ansätze speziell für Gebäudefassaden werden von Penard et al [2005] und Pollefeys et al. [2000] beschrieben. Bei der mobilen Erfassung kommen Laserscanner zum Einsatz, die auf Fahrzeugen montiert sind. Manandhar und Shibasaki [2002] zeigen eine Klassifizierung von Laserpunkten in Fassaden- und Nichtfassadenpunkte. Früh et al. [2005] kombinieren den Laserscanner mit einem Kamerasystem, um texturierte Polygon- bzw. Dreiecksnetze zu erzeugen. Die resultierenden Fassadenmodelle sind jedoch nicht interpretiert und somit unstrukturiert.

Gebäude und deren Fassaden weisen meist eine regelmäßige Gestaltung auf. Diese Eigenschaft machen sich beispielsweise generische Verfahren zunutze, um 3D-Gebäudemodelle mit detailliert ausgearbeiteten Fassaden automatisch zu erzeugen. Als Anwendung wird beispielsweise die Rekonstruktion einer archäologischen Mayastätte angeführt [Müller et al. 2006a]. Um die Gebäudeform und die Anordnung der Fassadenelemente zu beschreiben, verwenden Müller et al. [2006b] Grammatiken. Solche Regeln über den Aufbau einer Fassade können auch in den Rekonstruktionsprozess integriert werden. Ripperda und Brenner [2006] verwenden hierzu ebenfalls eine Grammatik, um die Menge der unterstützten Fassadenanordnungen allgemein zu beschreiben. In dem Lösungsraum, die sich aus den Ableitungsbäumen der Grammatik ergeben, ist eine gültige Rekonstruktion der gemessenen Fassade enthalten. Da dieser nicht vollständig nach einer möglichen Lösung durchsucht werden kann, werden über stochastische Methoden zufällige Hypothesen gebildet und mit gemessenen Laser- und Bilddaten der Fassade verglichen. Aufgebaut werden die Fassadenmuster mittels der reversible jump Markov Chain Monte Carlo Methode (jrMCMC) und einer Menge von Plausibilitätsfunktionen, die beispielsweise Annahmen über die Fenstergröße enthalten. Aus den sich ergebenden Ableitungsbäumen, den entsprechenden Parametervektoren und den Messwerten kann anschließend ein Wahrscheinlichkeitswert berechnet werden, für den es gilt ein Maximum zu erreichen. Die Schwierigkeit ist eine allgemeine Grammatik für Fassaden aufzubauen, die sowohl eine symmetrische und regelmäßige Struktur erzeugt, aber dennoch alle real existierende Konfigurationen zulässt. Ein ähnlicher Ansatz zur semantischen Interpretation von Fassadenbildern wurde auch schon von Alegre und Dallaert [2004] vorgestellt.

Mayer und Reznik [2006] interpretieren Gebäudefassaden aus Bildern mit dem Ziel Fenster zu erkennen. Dazu werden in den Bildern Fassadenregionen um markante Punkte, die mit dem Förstner-Operator detektiert wurden [Förstner und Gülch 1987], mit einer Referenzdatenbank verglichen. In dieser sind hunderte von Trainingsfenster enthalten, die zusätzlich zu den detektierten Punkten auch die Ausdehnung und die relative Position der Fenstermitte enthalten. Nach der positiven Identifikation einer Bildregion mittels des normalisierten Kreuzkorrelationskoeffizienten kann eine Hypothese über die Fensterposition aufgestellt werden. Der Einfluss falscher Hypothesen auf die resultierenden Positionen wird mit einem Gaußschen Filter eliminiert. Mittels der Markov Chain Monte Carlo Methode (MCMC) und Simulated Annealing wird anschließend iterativ die Ausdehnung der Fenster geschätzt. Die fehlenden Fenstertiefen liefert ein Plane Sweeping Ansatz bei dem die Fensterebene bis zu einer Maximaldistanz verschoben wird. Aus den Parametern der Fassaden-ebene und der Projektionsmatrize werden die Homographien als quadratische Grauwertdifferenzen zwischen Ebene und Bild berechnet. Die Entfernung mit der geringsten Differenz wird als Fensterebene angenommen.

2.5 CityGML

Die meisten 3D-Stadtmodelle werden bisher rein als graphische oder geometrische Modelle verwirklicht. Semantische Aspekte, d.h. beispielsweise die Klassifizierung von Objektteilen wie Fassade, Fenster, Türen, etc. und die Bereitstellung zusätzlicher thematischer Informationen, werden genau wie explizite Topologien, mit denen die relative Lage und Beziehungen der Teile zueinander ausgedrückt sind, häufig nicht berücksichtigt. Rein geometrische Modelle sind für Visualisierungsaufgaben ausreichend, nicht jedoch für komplexe Analysen oder thematische und räumliche Abfragen. Des Weiteren sind die Modelle in anwendungsspezifischen Datenformaten gespeichert und daher nur mit großem Aufwand in anderen Applikationen nutzbar. Um eine breitere Anwendung von 3D-Stadtmodellen zu ermöglichen, wurde von der Special Interest Group 3D (SIG 3D) der Initiative Geodateninfrastruktur Nordrhein-Westfalen (GDI NRW) das Austauschformat CityGML entwickelt (siehe [Kolbe et al. 2005]). Die über 70 Mitglieder der Interessensgemeinschaft stammen aus der Industrie, der öffentlichen Verwaltung und Forschungseinrichtungen der BRD und dem europäischen Ausland, die im Bereich der Entwicklung und kommerziellen Nutzung von interoperablen 3D-Stadtmodellen tätig sind. CityGML wird zurzeit international diskutiert und soll ein durch das Open Geospatial Consortium (OGC) anerkanntes Anwendungsschema für 3D-Stadtmodelle werden.

Das Datenmodell von CityGML basiert auf den ISO-Standards ISO 191xx und ist als Anwendungsschema der Geography Markup Language (GML3, siehe [Cox et al. 2004]) realisiert. GML3 wiederum setzt auf XML auf und erlaubt den Austausch raumbezogener Objekte mitsamt Attributen, Relationen und Geometrien. Es unterstützt einfache und komplexe 3D-Geometrien und deren Topologie und kann für spezifische Anwendungsgebiete erweitert werden. Als Multiskalen-Repräsentation konzipiert erlaubt CityGML die verlustfreie Speicherung und Übertragung der 3D-Geometrie, Topologie und Thematik von 3D-Stadtmodellen in fünf Detaillierungsgraden (LOD 0 bis LOD 4). Es wird zwischen Gelände, Gebäude, Vegetation, Gewässer und Verkehr unterschieden. Ein Datensatz kann dasselbe Objekt in mehreren Detaillierungsgraden gleichzeitig enthalten, aber auch Objekte mit unterschiedlichen Detaillierungsgraden in sich vereinigen. Das Herzstück von CityGML sind die Gebäudemodelle, deren Form, Genauigkeit und Mindestgröße durch die Detaillierungsgrade LOD 1 bis 4 definiert sind (siehe auch Tabelle 2-1 und Abbildung 2-10a-e):

	Beschreibung	Genauigkeit Lage	Genauigkeit Höhe	Mindestgröße Grundfläche
LOD 0	2.5D DGM mit überlagertem Luftbild oder Karte	–	–	–
LOD 1	Blockmodelle ohne Dachstrukturen oder Texturen	$\geq 5\text{m}$	$\geq 5\text{m}$	6m x 6m
LOD 2	Modelle mit Dachstrukturen und Texturen + Vegetationsobjekte	2m	1m	4m x 4m
LOD 3	Architekturmodelle mit detaillierten Fassaden- und Dachstrukturen (z.B. Balkone, Erker, Vorsprünge) mit hoch aufgelösten Texturen	0,5m	0,5m	2m x 2m
LOD 4	Begehbare Innenraummodelle (Räume, Türen, Treppen, Möbel usw.)	$\leq 0.2\text{m}$	$\leq 0.2\text{m}$	–

Tabelle 2-1: Detailstufen von CityGML mit Angabe von Genauigkeiten und Mindestgrößen.



Abbildung 2-10: 3D-Stadtmodelle in den Detailstufen 0 bis 4 (von links nach rechts) (aus [Kolbe 2006]).

Die Objekte werden in CityGML geometrisch mittels Randbeschreibung modelliert. Gebäudemodelle sind hierarchisch aus Gebäudeteilen aufgebaut, die einen Verweis auf thematische Daten in externen Datenbanken (z.B. Kataster, Facility Management Systeme, usw.) besitzen können. Zusätzlich enthalten Gebäudeteile die wichtigsten Sachattribute wie Name, Funktion, Baujahr, Dachtyp, Adresse, Anzahl ober- und unterirdischer Stockwerke, Geschosshöhen sowie für die Visualisierung Attribute über deren Erscheinungsbild. Ab LOD 2 können Flächen thematisch in Dach-, Wand- oder Bodenflächen unterschieden werden. Primitive wiederum lassen sich zu Flächen- und Körperaggregaten zusammenfassen, z.B. um Gebäudemerkmale wie Schornsteine, äußere Treppen, Balkone etc. dem Modell hinzuzufügen. Sowohl Primitive als auch Aggregate sind mithilfe der GML3 Geometrie- und Topologieklassen implementiert. Etliche Integritätsbedingungen garantieren eine saubere Topologie ohne Redundanzen. Ab LOD 3 können Öffnungen wie Türen und Fenster modelliert und klassifiziert werden. LOD 4 vervollständigt das Modell durch die Hinzunahme interner Raumstrukturen mitsamt deren Möblierung.

Bisher wurde die allgemeine Erfassung und Modellierung von 3D-Stadtmodellen behandelt, wie sie als Detailstufen LOD 1 bis LOD 3 von CityGML definiert werden. Die vorherrschende Datengrundlage von Städten und Kommunen sind Gebäudemodelle mit detaillierten Grundrissen und Dachstrukturen (LOD 2), jedoch ohne eine flächendeckende Verfügbarkeit von Fassaden- und Dachtexturen. In den folgenden Kapiteln soll nun die Aufbereitung dieser Daten thematisiert werden. Das betrifft vor allem die Generalisierung der Modelle, d.h. die Vereinfachung der Geometrie unter Beachtung kartographischer Aspekte.

3 Generalisierung

Laut Hake et al. [2002] wird in der klassischen Kartographie die Notwendigkeit der Generalisierung mit dem Grundkarte-Folgekarte-Prinzip begründet. Die Realität wird entsprechend dem Umfang und Genauigkeit der erfassten Daten in ein digitales Objektmodell oder eine Grundkarte umgesetzt. Neben der Aufbereitung sachlicher und statistischer Daten findet dabei eine messtechnische Vereinfachung statt, wodurch die resultierenden Daten im Vergleich zur Realität in einem erheblichen Verkleinerungsverhältnis stehen. Dies wird auch als Erfassungsgeneralisierung bezeichnet. Vor der eigentlichen Kartenerstellung kann eine vorbereitende Modellgeneralisierung erfolgen, welche einen reduzierten Datensatz nach thematischen Kriterien für eine sich anschließende Analyse oder Präsentation erzeugt. Sester [2000] spricht in diesem Zusammenhang auch von einer Datenbankgeneralisierung bzw. Datenbankabstraktion. Aus den Grundkarten lassen sich durch simples Verkleinern jedoch keine Folgekarten erzeugen, da Objekte unterhalb des Betrags der graphischen Mindestgröße unleserlich wiedergegeben würden. Vielmehr ist das Prinzip der Lesbarkeit zu beachten, so dass Objekte aufgrund ihrer Bedeutung vergrößert oder aus Mangel an Darstellungsfläche erst gar nicht wiederzugeben sind. Solche Operationen schränken allerdings das Prinzip der geometrischen Richtigkeit bzw. das Prinzip der Vollständigkeit ein. Die kartographische Generalisierung bedient sich elementarer geometrischer und sachlicher Vorgänge, um Folgekarten unmittelbar aus anderen Karten und unter Berücksichtigung der genannten Prinzipien abzuleiten. Folgende elementaren Vorgänge zur kartographischen Generalisierung werden in der Literatur aufgelistet (vgl. [Hake et al. 2002], [Meng 1997] und [Sester 2000]):

1. Vereinfachen (Glätten) der geometrischen Form unter Wahrung der Objektcharakteristika.
2. Vergrößern von aufgrund des Kartenmaßstabs nicht mehr erkennbarer Objekte.
3. Verdrängen um räumliche Konflikte zwischen Objekten zu beheben.
4. Zusammenfassen (Aggregation) benachbarter Objekte.
5. Auswählen (Selektion), d.h. Erhalten oder Fortlassen entsprechend der relativen Bedeutung der Objekte.
6. Klassifizieren aufgrund gemeinsamer Attribute. Das Objekt erhält in Anhängigkeit von der aktualisierten Objektbeschreibung eine neue Repräsentation.
7. Bewerten, indem Objekte gemäß ihrer relativen Bedeutung geometrisch betont oder gemindert werden.
8. Typifizieren, d.h. Ausdünnen einer Menge gleichartiger Objekte unter Beibehaltung der ursprünglichen Objektverteilung.
9. Kollaps durch Reduktion der Dimension von Objekten.

Diese Einzeloperatoren werden unter Beachtung von graphischen, geometrischen und strukturellen Bedingungen in der kartographischen Generalisierung sinnvoll verknüpft. In der vorliegenden Arbeit wird ausschließlich auf das geometrische Vereinfachen von einzelnen 3D-Gebäudemodellen unter Wahrung der objektspezifischen Formcharakteristika eingegangen. Dabei sind vor allem die geometrischen Bedingungen Parallelität, Geradlinigkeit und Rechtwinkligkeit von Gebäudeflächen von Bedeutung. Eine maßstabsbedingte Mindestgröße von Strecken und Flächen soll der Vereinfachungsoperation als graphische Bedingung vorgegeben sein.

Der Rest des Kapitels widmet sich dem Stand der Forschung, der sich aus der Generalisierung von 2D- und 3D-Modellen von einzelnen Gebäuden (Abschnitt 3.1 und 3.2) und der Flächensimplifizierung allgemeiner 3D-Körper (Abschnitt 3.3) zusammensetzt.

das Muster mit der höchsten Korrelation ersetzt. Ein der Musterklasse Sonderform zugeordneter Gebäudegrundriss wird dabei mit dem Generalisierungsansatz von Staufenberg [1973] vereinfacht.

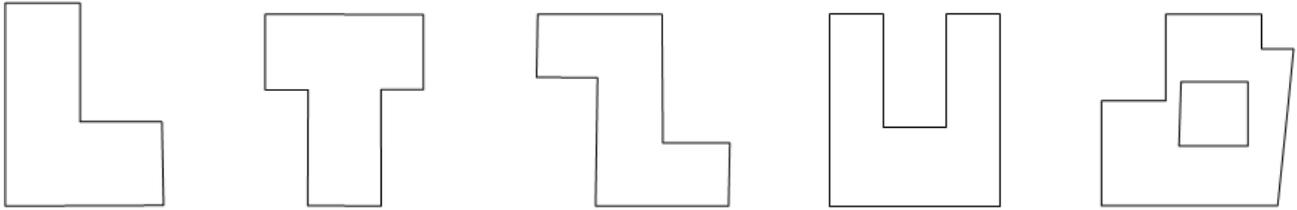


Abbildung 3-2: Die Musterklassen L, T, Z, U und Sonderform (aus [Meyer 1989]).

Ein sehr ähnlicher Ansatz wird in [Rainsford und Mackaness 2002] dargestellt. Das Verfahren automatisiert die Formvereinfachung von freistehenden Bauernhöfen in Dänemark. Der Katalog der Dänischen Landesvermessungsbehörde (KMS) umfasst die an Buchstaben des Alphabets angelehnten Muster I, F, P, G, E, L, U, O und T. Ursprünglich wurde jeder Grundriss manuell einem Muster zugeordnet und durch eine entsprechende Vektordarstellung ersetzt. Das vorgestellte Verfahren automatisiert diesen Prozess, indem offene Ringe des Grundrisspolygons zunächst geschlossen werden. Ein Grundriss gehört dann entweder zur Gruppe mit einem Ring (I, F, E, L, U, T), zwei Ringen (P, G, O) oder kann keinem Muster zugeordnet werden. Anschließend wird das Grundrisspolygon orthogonalisiert und kleine Fortsätze iterativ entfernt. Nach jeder Iteration wird das Polygon in eine Winkelsequenz umgewandelt. Ein 90° Winkel wird dabei mit einem L für eine Linkskurve und ein 270° Winkel mit einem R für eine Rechtskurve kodiert. Für das Muster F würde man beispielsweise die Winkelsequenz RRLLRRLRR erhalten. Die abgeleitete Buchstabenfolge wird mit dem Musterkatalog verglichen, wobei die Sequenz rotiert und gespiegelt wird (siehe Abbildung 3-3). Der Algorithmus endet, sobald ein Muster identifiziert wurde und das Objekt wird durch das Muster ersetzt.

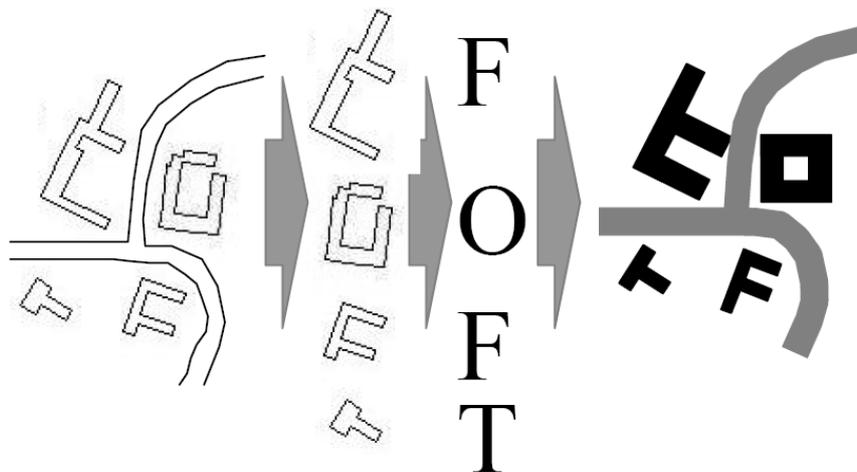


Abbildung 3-3: Generalisierung mit Mustern nach [Rainsford und Mackaness 2002].

Li [1996] wendet für 2D-Rasterdaten Opening und Closing an, um einfache Strukturelemente mit glatten Konturen zu vereinfachen. Su et al. [1997] erweitern diesen Ansatz. Der Fokus der Arbeit liegt in der Aggregation von Gebäudegrundflächen mittels morphologischer Operatoren. Die Kontur der zusammengefassten Gebäude wird anschließend durch die Bildung der konvexen Hülle und das Filtern von Unregelmäßigkeiten vereinfacht. Urena Camara und Ariza Lopez [2005] verwenden unter anderem morphologische Operatoren, um großflächige Gebäudeblöcke von Stadtkarten zusammenzufassen und zu vereinfachen.

Ein Ansatz, welcher die rasterbasierten Verfahren auf den Vektorraum erweitert, ist in der Arbeit von Mayer [1998] dargestellt. Er bedient sich der aus der Bildverarbeitung bekannten Maßstabsraumtheorie (Scale-Space Theory). Diese befasst sich mit der formalen Behandlung von Rasterbildstrukturen in unterschiedlichen Maßstäben [Lindeberg 1994]. Aus einer gegebenen Repräsentation wird durch sukzessive Unterdrückung kleiner Merkmale eine Reihe weiterer Repräsentationen in unterschiedlichen Maßstäben abgeleitet.

Die sich daraus ergebenden Maßstabsraum-Repräsentationen lassen sich über einen Maßstabsparameter beschreiben. In der Bildanalyse werden die Objektstrukturen dann entweder im geeigneten Maßstab oder über verschiedene Maßstäbe hinweg gleichzeitig analysiert. Ein Beispiel für die Definition eines Maßstabsraums ist die mathematische Morphologie.

Im dargestellten Verfahren werden die morphologischen Operatoren Opening und Closing auf die Vektordarstellung des Gebäudegrundrisses angewendet. Dies geschieht über die Verschiebung aller Kanten in oder gegen die Richtung ihrer Normalen. Gebäude werden dadurch zusammengefasst oder schmale Verbindungen zwischen Gebäudeteilen gelöscht. Ein weiterer Maßstabsraum ist der diskrete Krümmungsraum. In diesem werden ausgewählte Kanten, die unter einem diskreten Längenschwellwert liegen und einen kleinen Krümmungsradius besitzen, verschoben (siehe Abbildung 3-4). Die Bewegungsrichtung der Kanten kann im Gegensatz zu Opening und Closing in verschiedene Richtungen erfolgen.

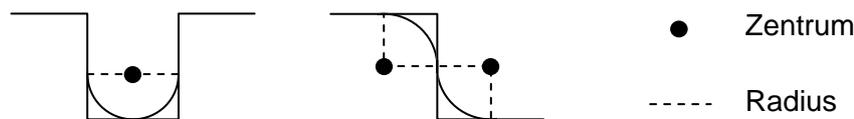


Abbildung 3-4: Diskreter Krümmungsraum mit Zentrum und Radius (nach [Forberg 2005]).

Unterschieden werden U-, Z- und L-Strukturen, für die es jeweils eine unterschiedliche Vereinfachungsstrategie gibt (siehe Abbildung 3-5). Die kurze Kante einer U-Struktur wird so verschoben, dass sich die übergeordnete Struktur schließt. Für Z-Strukturen werden die langen Kanten gegenläufig bewegt, bis diese sich treffen. Die kurze Kante dazwischen wird dabei eliminiert. Die zwei kurzen Kanten einer L-Struktur werden gleichzeitig nach innen oder außen verschoben, bis sie sich im neuen Eckpunkt treffen.

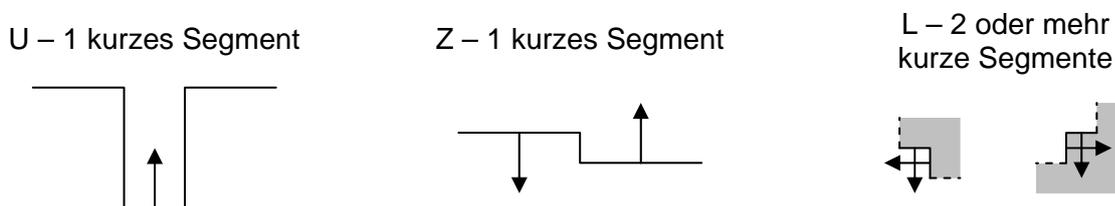


Abbildung 3-5: Strukturen des diskreten Krümmungsraums (nach [Forberg 2005]).

Die vorgestellten Operatoren sind auf orthogonal angeordnete Kanten angewiesen. Nahezu orthogonale Formen können zwar orthogonalisiert werden. Trotzdem ist es wegen dieser Einschränkung nicht möglich mit dem Verfahren beliebige Grundrisse zu vereinfachen.

Ähnlich dem Ansatz von Staufenbiel [1973] führt Sester [2000] eine Regelmenge zur Elimination von Kleinteilen ein. Es können drei Fälle auftreten, die über die zwei Richtungen der Kanten vor und nach der Kleinteile unterschieden werden (vgl. Abbildung 3-6).

1. Bei einem Versatz haben beide Kanten dieselbe Richtung. Die längere Kante wird verlängert und mit der entsprechenden Folgekante geschnitten.
2. Zeigen Vorgänger- und Nachfolgerkante in unterschiedliche Richtungen, so handelt es sich um eine Ein- oder Ausbuchtung. Hier wird je nach Fall die entsprechende Folgekante verlängert und mit einer Seitenkante der Ein- oder Ausbuchtung verschnitten.
3. Bilden Vorgänger- und Nachfolgerkante ungefähr einen rechten Winkel, so werden beide Kanten verschnitten und die Ecke entfällt.

Durch iterative Anwendung der drei Regeln entsteht zunächst eine Näherungslösung, die anschließend mittels einer Ausgleichung nach kleinsten Quadraten an den ursprünglichen Gebäudegrundriss angepasst wird. Dazu wird das Modell in eine parametrische Darstellung überführt, welche das Gebäude als Funktion von

Breiten und Längen entlang der zwei Hauptachsen beschreibt (siehe Abbildung 3-7). Bei der Formoptimierung gleichen sich die Kanten an die Hauptachsen an. Bei rechtwinkligen Hauptachsen wird die Rechtwinkligkeit im ausgeglichenen Modell für nahezu rechtwinklige Kanten erzwungen. Des Weiteren können durch den Ausgleichsprozess bestimmte Eigenschaften beeinflusst werden, so dass beispielsweise eine Betonung wichtiger, gestreckter Gebäudeteile erfolgt.

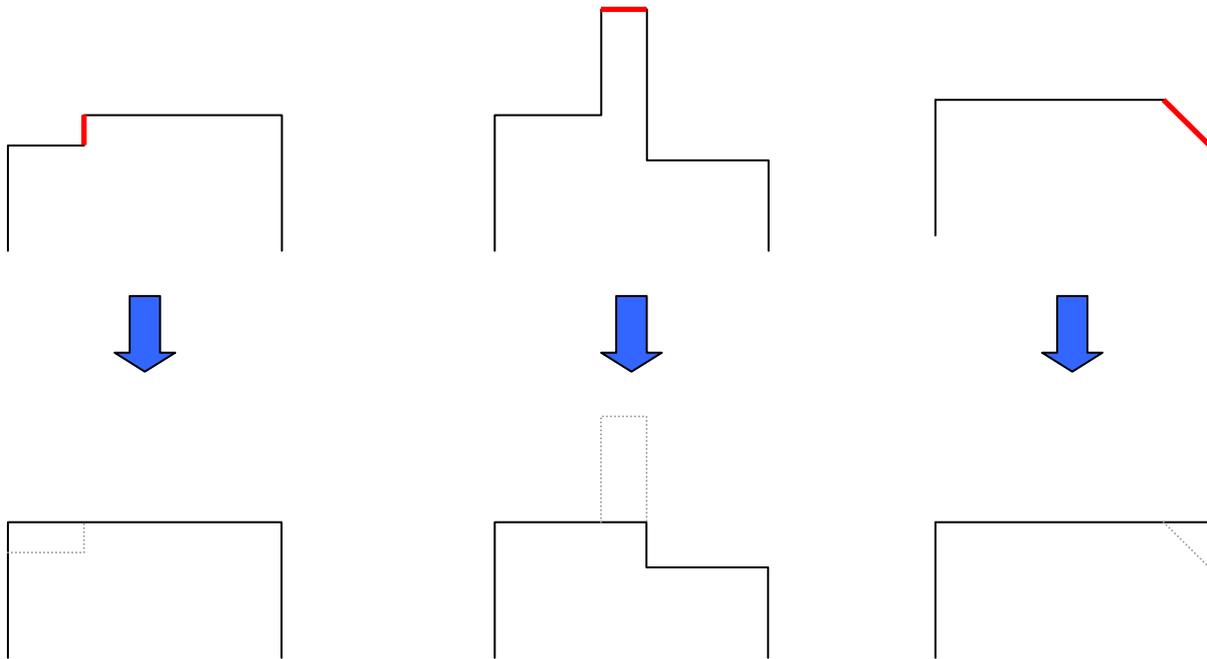


Abbildung 3-6: Elimination zu kurzer Kanten (rot) bei Versatz, Ausbuchtung und Ecke (nach [Sester 2000]).

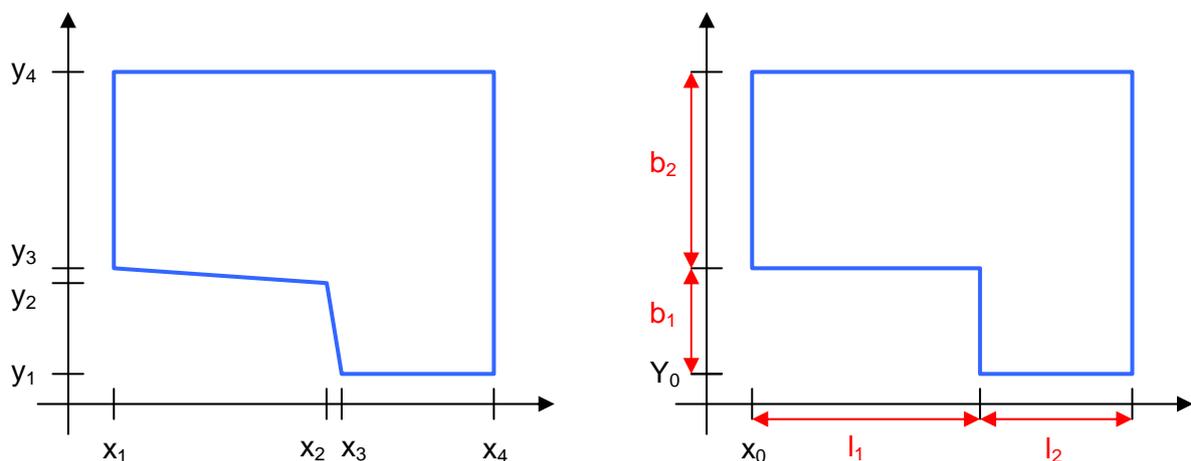


Abbildung 3-7: Bestimmung und Ausgleichung der Formparameter in Abhängigkeit von den Hauptachsen (nach [Sester 2000]).

Für dichte Bebauung, z.B. in Innenstadtgebieten, wird die gemeinsame Vereinfachung benachbarter Objekte vorgeschlagen. Erreicht wird das durch Zusammenfassen von Gebäuden mittels der morphologischen Operation Closing. Anschließend können die aggregierten Gebäude als Einzelobjekte vereinfacht werden.

Basierend auf der Idee der progressiven Speicherung von Dreiecksnetzen [Hoppe 1996] erweitern Brenner und Sester [2003] den Ansatz um eine sich kontinuierlich dem Maßstab angleichende Darstellung für kleine Bildschirme. Es werden Basisoperationen (Simple Operations) zum Einfügen, Verdoppeln, Bewegen und Entfernen von Knoten definiert. Diese werden wiederum zu elementaren Generalisierungsoperationen kombiniert. Für jede Kombination von einfachen Operationen kann nun eine weitere Kombination zur Umkehrung der Generalisierung definiert werden. Während des Generalisierungsprozesses werden die verwendeten Generalisierungsoperationen mitsamt den Parametern in einer Generalisierungskette gespeichert, die auch wieder umgekehrt werden kann. Dadurch ist es möglich aus einem generalisierten Gebäudegrundriss wieder das Ursprungsgebäude mitsamt den Zwischenrepräsentationen zu erhalten (vgl. Abbildung 3-8). Durch die Einführung der Operation 'Knoten bewegen', kann beim Zoomen ein fließender Übergang zwischen den Detailstufen erreicht werden, indem der Knoten in einer Animation auf seine neue Position geschoben wird. Sester und Brenner [2004] zeigen wie eine kontinuierliche Generalisierung auch für die Typifizierung und Verdrängung von Gebäuden realisiert werden kann.

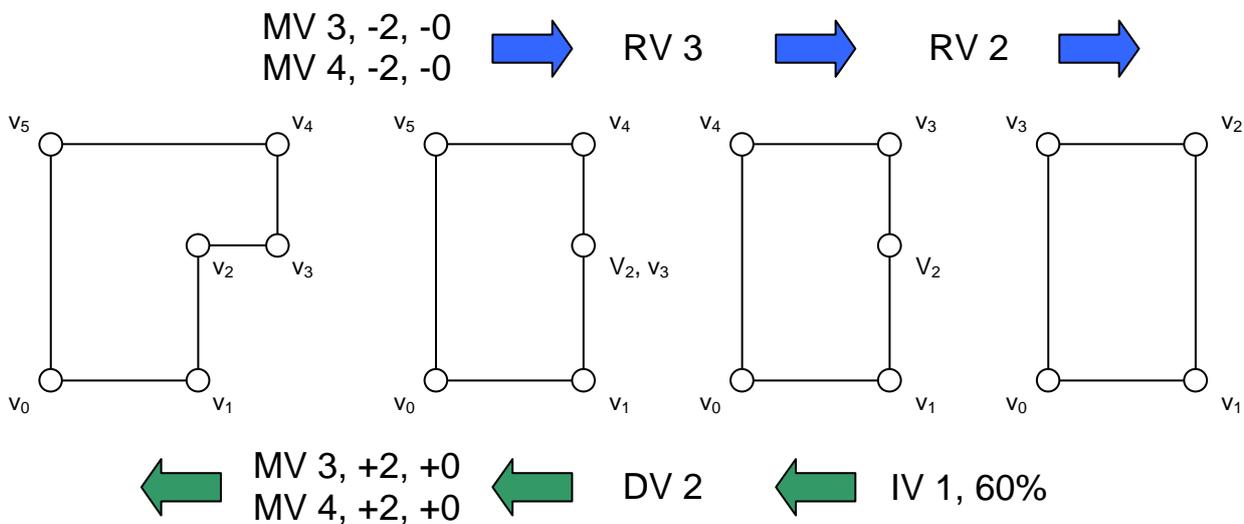


Abbildung 3-8: Generalisierung eines L-förmigen Gebäudegrundrisses (blau) und entsprechende Umkehrung (grün) mittels der Basisoperationen MV (Move Vertex), RV (Remove Vertex), DV (Duplicate Vertex) und IV (Insert Vertex).

Um das beschriebene Verfahren anzuwenden zu können, müssen vorher die Hauptachsen bekannt sein. Eine Aufgabe, die für beliebige Formen nicht immer trivial ist. Duchene et al. [2003] beschreiben verschiedene Maßregeln für die Gebäudeorientierung. Hierbei wird unterschieden zwischen der allgemeinen Orientierung und der Orientierung der Wände. Die Achsen des kleinsten Minimalumschließenden Rechtecks wird als das geeignete Maß für die allgemeine Orientierung identifiziert. Für die Orientierung der Wände wurde ein neues Maß eingeführt, welches die häufigste Orientierung statistisch gewichtet. Über ein komplexes Regelwerk, welches die gefundenen Orientierungen vergleicht, wird dann die Gebäudeorientierung bestimmt. Des Weiteren wird untersucht, wie sich die Form der Gebäude nach den Mustern von Meyer [1989] zusätzlich nutzen lassen.

Die Verfahren zur Generalisierung von 2D-Gebäudegrundrissen basieren im Wesentlichen auf der Erkennung und Entfernung von Merkmalen bzw. Kleinseiten mittels expliziter Regeln, der vektoriellen Mustererkennung oder der morphologischen Operatoren. All diese Konzepte finden sich in einer mehr oder weniger abgewandelten Form in der Vereinfachung von 3D-Gebäudemodellen wieder (siehe Abschnitt 3.2 und Kapitel 4). Deren Anpassung an die Verarbeitung von 3D-Modellen ist jedoch äußerst schwierig. Im Vergleich zum Liniensegment, welches genau einen Vorgänger und einen Nachfolger besitzt, kann ein Polygon an beliebig viele Nachbarflächen grenzen. Des Weiteren gelten für die Dachstrukturen von 3D-Gebäudemodellen nicht dieselben geometrischen Relationen wie für die Grundrisspolygone. Hier sind andere Methoden zu entwickeln, um Dachsymmetrien zu bewahren (siehe Kapitel 5).

Für die Erzeugung von Blockmodellen, LOD 1 in CityGML, ist allerdings keine echte 3D-Generalisierung notwendig. Hierfür sind die bekannten Verfahren zur Grundrissgeneralisierung ausreichend. Ein Blockgebäude kann durch Extrusion des zweidimensional generalisierten Gebäudegrundrisses auf eine gemittelte Dachhöhe generiert werden. Wie man in Kapitel 5 sehen wird, ist eine Aufteilung der 3D-Generalisierung in Grundrissgeneralisierung und die Generierung einer vereinfachten Dachstruktur möglich.

Das Konzept, die vereinfachte Form an die Originalgeometrie anzupassen, wird in Kapitel 4 bei der Generalisierung von 3D-Gebäudemodellen wieder aufgegriffen. Die Lage und Orientierung der Flächen von in Randbeschreibung gegebenen Gebäudemodellen werden unter Berücksichtigung der geometrischen Relationen optimiert.

3.2 Generalisierung von 3D-Gebäudemodellen

Obwohl die Generalisierung von 3D-Gebäudemodellen ein junges Forschungsgebiet ist, welches erst mit der Verfügbarkeit von automatisch rekonstruierten 3D-Stadtmodellen ein bedeutsames Interesse erhielt, existieren mittlerweile einige Verfahren zur geometrischen Vereinfachung von Einzelgebäuden. Zumeist handelt es sich dabei um Erweiterungen von Algorithmen, die für die Generalisierung von 2D-Grundrisspolygonen entwickelt wurden. Die in Kapitel 4 und Kapitel 5 vorgestellten Arbeiten wurden erstmals in [Kada 2002] und [Kada 2005] veröffentlicht und sind demnach zur selben Zeit entstanden wie die meisten nachfolgend vorgestellten Verfahren.

Basierend auf der 2D-Generalisierungssoftware CHANGE [Powitz 1993] stellt Anders [2005] eine Methode zur Aggregation und Formvereinfachung von 3D-Gebäudegruppen vor. CHANGE vereinfacht und verbindet 2D-Gebäudegrundrisse für topographische Karten im Maßstab 1:5.000 und 1:25.000. Die architektonische Struktur wird dabei weitestgehend bewahrt. Die Software wird für linear angeordnete 3D-Gebäudegruppen, die in der Darstellung nicht mehr als einzelne Objekte zu erkennen sind, wie folgt eingesetzt: Ein Minimalumschließender Quader wird berechnet, um aus der Länge, Breite und Höhe drei Projektionsrichtungen abzuleiten. Durch orthographische Projektion werden anschließend drei 2D-Ansichten der Gebäudegruppe generiert und mittels CHANGE generalisiert (siehe Abbildung 3-9).

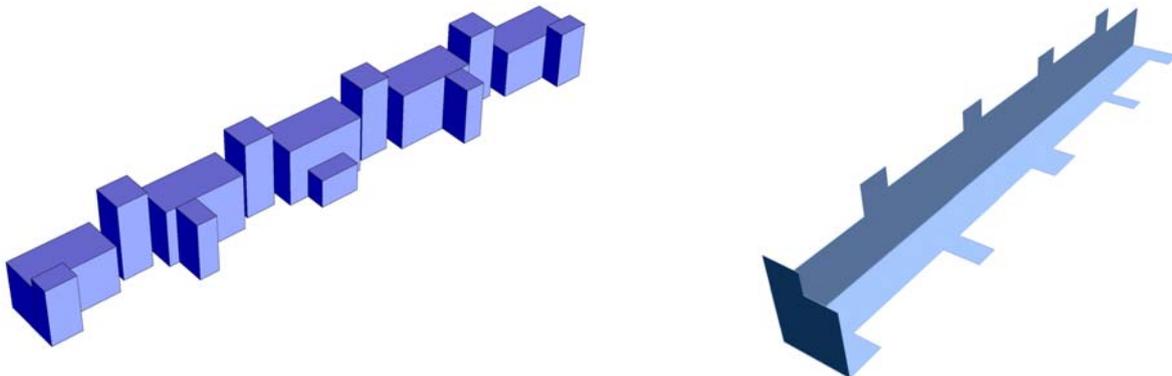


Abbildung 3-9: Gruppe von 3D-Gebäudemodellen (links) und generalisierte Projektionen (rechts).

Die vereinfachten 2D-Polygone werden extrudiert und die daraus entstandenen drei Festkörpergeometrien mittels einer booleschen Verschneidungsoperation zum endgültigen 3D-Gebäudemodell vereinigt (siehe Abbildung 3-10). Dieser Ansatz reduziert sowohl die Anzahl der Gebäudeflächen als auch die Gebäudeobjekte, liefert jedoch für asymmetrische Gebäudegruppen keine sehr guten Ergebnisse. Wobei das Ergebnis von den gefundenen Projektionsrichtungen abhängig ist. Asymmetrische Strukturen gehen in diesem Ansatz grundsätzlich verloren. Es können bisher nur flache Dachstrukturen behandelt werden.

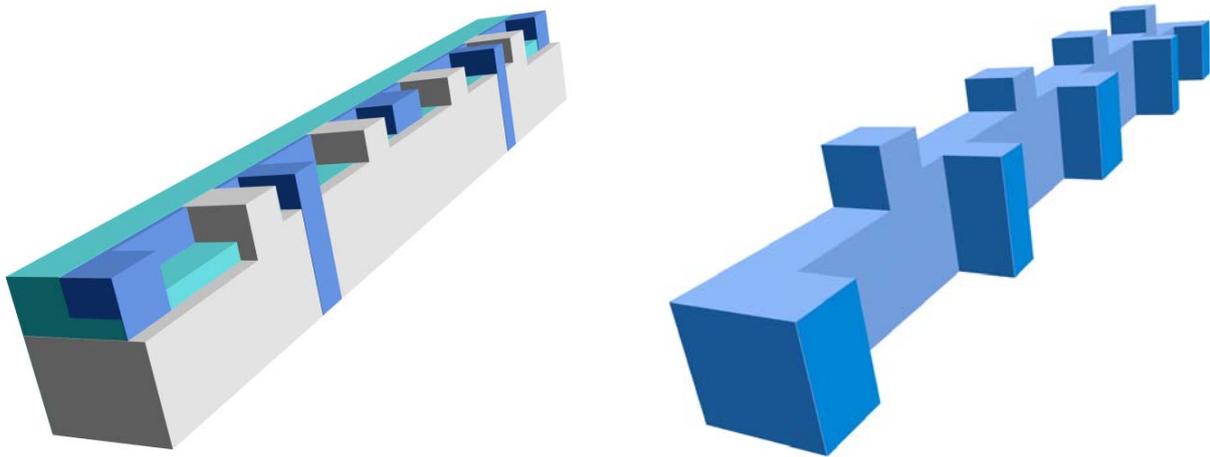


Abbildung 3-10: 3D-Extrusion der generalisierten Projektionen (links) und das resultierende Gebäudemodell nach der Verschneidung der extrudierten Projektionen (rechts) (aus [Anders 2005]).

Ein frühes Konzept zur Generalisierung von 3D-Gebäudemodellen wird in [Sester und Klein 1999] vorgeschlagen. Mittels eines objektabhängigen Regelwerks werden Gebäudeteile entsprechend ihrer Bedeutung und Wichtigkeit behandelt. Zum Beispiel werden nahe beieinander liegende Fenster zusammengefasst, Objekte je nach Bedeutung vergrößert, weggelassen oder verschoben und Grundrisse vereinfacht (Abbildung 3-11). Dies ist jedoch nur möglich, wenn die Gebäudemodelle semantisch strukturiert sind, d.h. wenn sie aus mehreren getrennten Objektteilen zusammengesetzt sind. Für automatisch erfasste Gebäudemodelle trifft dies in der Regel jedoch nicht zu. Vor der Generalisierung müssen die Modelle daher zunächst in ihre Bestandteile zerlegt werden, für die es anschließend gilt, die entsprechenden Objektklassen zu identifizieren.

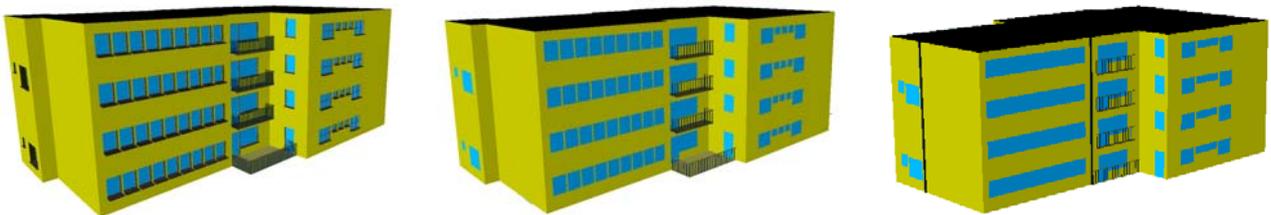


Abbildung 3-11: Ein 3D-Gebäudemodell in drei Detailstufen (aus [Sester und Klein 1999]).

Ein Konzept zur Zerlegung von 3D-Gebäudemodellen wird von Thiemann [2002] vorgestellt. Er definiert Bestandteile als 3D-Primitive, welche aus dem Hauptkörper herausstehen, Löcher formen oder Schnittkanten auffüllen. Die Segmentierung basiert auf dem in Abschnitt 3.3 beschriebenen Verfahren zur Erkennung und Entfernung von Merkmalen [Ribelles et al. 2001]. Durch die Anwendung heuristischer Regeln können Thiemann und Sester [2004] die Anzahl der zur Segmentierung notwendigen Schnittebenen reduzieren und damit die Zeitkomplexität des Verfahrens. Die 3D-Primitive werden mit möglichst einfachen Schnittkombinationen vom Hauptkörper getrennt und hierarchisch über Booleschen Operatoren in einem CSG-Baum organisiert. Nach der Segmentierung erfolgt eine Interpretation der Primitive, bei der die geometrischen Attribute Größe, Form, Position, Ausrichtung, topologische Relation, Konnektivität usw. in Relation zum Hauptkörper betrachtet werden [Thieman und Sester 2005]. So können die für Gebäude üblichen Bestandteile wie Fenster, Türen, Balkone, Dachüberstände, Schornsteine, Dachluken und Gauben identifiziert werden. Das 3D-Gebäudemodell in Abbildung 3-12 wurde beispielsweise mit Hilfe von 34 Schnittebenen segmentiert. Daraus resultieren 19 Bestandteile, die an Hand ihrer Klassifikation farblich gekennzeichnet sind. Eine auf diese Weise aufgebaute Generalisierungshierarchie kann nun gemäß der Bedeutung bzw. Semantik ihrer Bestandteile mittels der Generalisierungsoperationen Weglassen, Verdrängen, Betonen, Zusammenfassen und Typifizieren vereinfacht werden.

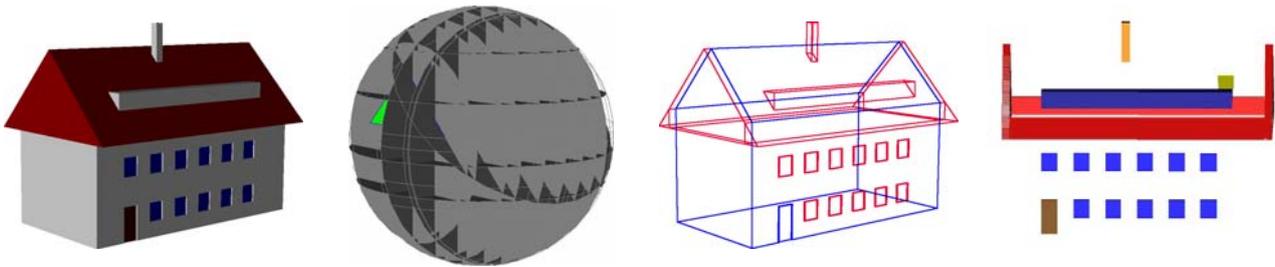


Abbildung 3-12: Das 3D-Gebäudemodell wird mittels 34 Segmentierungsebenen in ein aus 19 Teilen bestehendes Zellenmodell überführt. Die Bestandteile sind gemäß ihrer Semantik farblich gekennzeichnet. (aus [Thiemann und Sester 2004] und [Thiemann und Sester 2005])

Um kontinuierliche Übergänge zwischen unterschiedlichen Detailstufen zu ermöglichen, schlägt Thiemann [2003] eine progressive Speicherung der Generalisierungshierarchie vor. Aufbauend auf dem Hauptkörper werden ausschließlich die Differenzen zwischen den Detailstufen und deren Boolesche Verknüpfungen zueinander gespeichert. Dies erfolgt entweder in umgekehrter Reihenfolge zur Segmentierung oder nach einer Reorganisation des Graphen unterteilt in semantische Gruppen. So können unabhängig von der Objektgröße die bedeutsamen Bestandteile früher zum Hauptkörper hinzugefügt werden als die unwichtigen. Der Wechsel zwischen den Detailstufen erfolgt durch die Generalisierungsoperationen in der CSG-Repräsentation. Für die Darstellung wird das Modell anschließend in eine spezielle Randbeschreibung konvertiert, welche die charakteristischen Gebäudeeigenschaften, beispielsweise die Parallelität und Rechtwinkligkeit von Wänden, als Bedingungen implizit enthält [Thiemann 2001]. Eine visuelle Glättung beim Übergang von einem Maßstab zum nächsten wird mittels Geomorphing oder Fading berücksichtigt.

Stüber [2006] beschreibt ein nahezu identisches Verfahren zur geometrischen Vereinfachung von einzelnen 3D-Gebäudemodellen. Schwerpunkt der Arbeit ist die kartenverwandte (Echtzeit-) Darstellung von Stadtmodellen unter perspektivischer Zentralprojektion. Hierzu werden kartographisch geprägte Mindestgrößen für die Bildschirmdarstellung untersucht. Aus diesen lassen sich Sichtbarkeitsintervalle für die Gebäudebestandteile in der Generalisierungshierarchie ableiten, so dass Gebäudedetails in Abhängigkeit von ihrer Größe und von der Entfernung zum Betrachter ein- und ausgeschaltet werden können. Des Weiteren erfolgt auf Basis dieser Informationen eine Aggregation von Objekten, durch die angrenzende Gebäude bzw. Gebäudeteile mit vergleichbarem Sichtbarkeitsintervall zusammengefasst werden. In einem iterativen Prozess werden beispielsweise zuerst die zwei gleichgroßen Garagen, die sich zwischen zwei Gebäuden befinden, zusammengefasst. Erst wenn der Betrachter sich weit genug entfernt, so dass die Gesamtheit der beiden Garagen nicht mehr sichtbar ist, werden diese entfernt und die beiden Gebäude wachsen zusammen. Zur Generalisierung von Texturen werden in der Arbeit Resamplingverfahren aus der digitalen Bildverarbeitung (z.B. bilineare Interpolation) und die notwendige Texturgröße in Abhängigkeit von der Betrachtungsentfernung diskutiert. Die Schlussfolgerung des Autors legt nahe, dass eine weiterführende Generalisierung nicht notwendig ist. Für die Verwaltung der hierarchisch aufgebauten 3D-Gebäudemodelle mitsamt deren Aggregats- und Sichtbarkeitsinformationen wird des Weiteren ein Datenmodell für multiskalige 3D-Stadtmodelle vorgestellt.

Mit starkem Fokus auf die Generalisierung von 3D-Landmarken schlagen Thiemann und Sester [2006] die Verwendung von 3D-Mustern vor. Der Ansatz basiert auf der Annahme, dass sich Gebäudeformen durch eine geringe Anzahl von Klassen kategorisieren lassen. Ist die Gebäudeklasse bekannt, so lässt sich das Objekt durch eine Instanz des entsprechenden Musters ersetzen. Im Gegensatz zum Verfahren von Raisford und Mackaness [2002] werden die Muster jedoch nicht nur einfach gedreht und skaliert, sondern jedes verfügt über einen individuellen Parametersatz. Das Muster einer Kirche kann beispielsweise Parameter für die Anzahl, Lage und Form der Türme enthalten, die für das charakteristische Erscheinungsbild bestimmter Kirchenobjekte entscheidend sind. Die Herausforderung des Ansatzes liegt in der Definition solcher generischer Muster, der Klassifizierung von Gebäudemodellen und der Anpassung der erzeugten Instanzen an die originalen Gebäudeformen. In der Arbeit wird vornehmlich auf den letzten Punkt eingegangen. Für die Anpassung wird die Gebäudeinstanz von ihrer Parameterform in eine spezielle Randbeschreibung umgewandelt, bei der die polygonalen Flächen entlang der Normalenrichtungen frei beweglich sind. Über einen Optimierungsprozess werden die Flächen solange verschoben bis sich das vereinfachte Modell bestmöglich dem Originalgebäude annähert. Bis zu einem gewissen Grad kann sich dabei die Form und Topologie ändern, so dass beispielsweise aus dem Muster eines Walmdachs eine Gebäudeinstanz mit Giebeldach entsteht. Der

gesamte Vorgang wird über eine Metrik gesteuert, welche die Distanz zwischen einer auf dem Originalmodell gleichmäßig verteilten Punktmenge und den Flächen des vereinfachten Polygonmodells beschreibt. Da funktionale Abhängigkeiten zwischen den Flächen und den korrespondierenden Punkten bestehen, wird die optimale Form des vereinfachten 3D-Gebäudemodells über eine Ausgleichung nach kleinsten Quadraten gefunden. Abbildung 3-13 zeigt das Ergebnis von zwei generalisierten Gebäudemodellen. Unklar bei dem Verfahren ist bisher, wie einfach sich ein beliebig geformtes Modell einer Gebäudeklasse zuordnen lässt. Sobald jedoch ein geeignetes Muster gefunden wurde, kann zusätzlich zur Formvereinfachung auch eine Betonung der charakteristischen Gebäudemerkmale vorgenommen werden. Für das Modell einer Kirche kann dies beispielsweise die Vergrößerung der Kirchtürme bedeuten, so dass diese und damit auch das Gesamtmodell noch aus großer Distanz gut erkennbar sind.

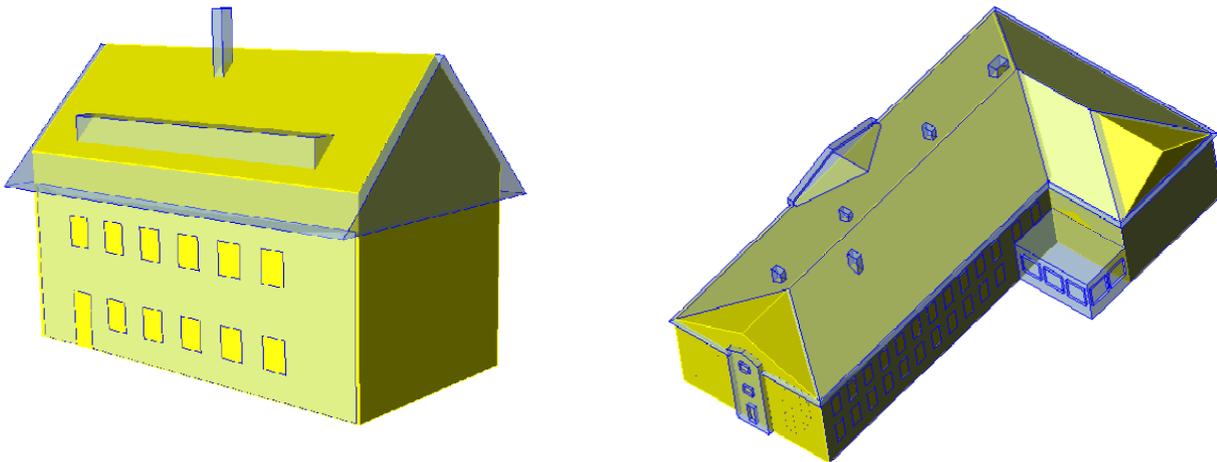


Abbildung 3-13: Generalisierung eines 3D-Gebäudemodells mit Giebeldach (links) und mit komplexer Dachstruktur (rechts). Dem Ergebnis (gelb) wurden die Originalmodelle (blau) überlagert (aus [Thiemann und Sester 2006]).

Mayer [2000] stellt eine Erweiterung der Generalisierung mit Maßstabsräumen (siehe Abschnitt 3.1) für die Anwendung auf 3D-Gebäudedaten vor. Das Ziel ist die automatische Ableitung von Modellen in verschiedenen Detailstufen. Es kommen dreidimensionale Ausführungen der morphologischen Operatoren Opening und Closing für Vektordaten zum Einsatz um Zwischenräume zu füllen oder Objekte zu trennen (siehe Abbildung 3-14).

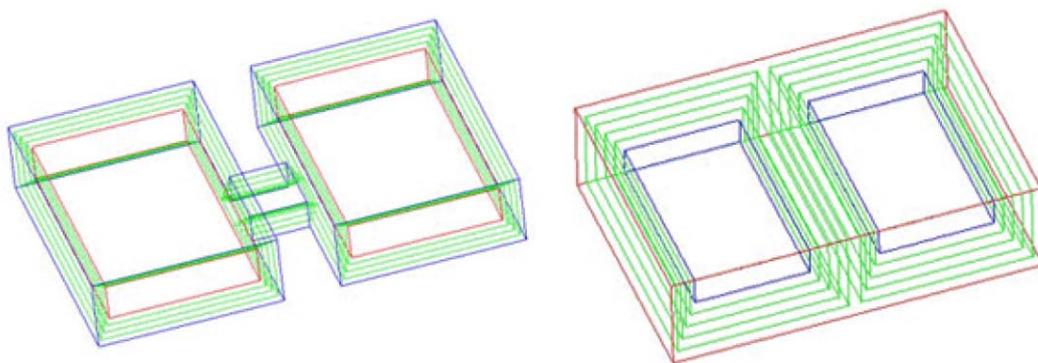


Abbildung 3-14: Aufspaltung mittels Erosion (links) und Verschmelzung mittels Dilatation im 3D-Raum (rechts): blau – Ausgangsobjekt, grün – inkrementelle Schritte, rot – resultierendes Objekt (aus [Forberg 2005]).

Auch die Operatoren im diskreten und kontinuierlichen Krümmungsraum werden auf die dritte Dimension erweitert um beispielsweise stufenförmige Objektdetails zu eliminieren. Im Gegensatz zu Opening und Closing werden parallele Flächen dabei in verschiedene Richtungen bewegt. Die vorgestellten Operatoren arbei-

ten allerdings nur auf rechtwinkligen Strukturen. Gebäudemodelle mit Giebel- oder Walmdach können mit diesen Operatoren nicht verarbeitet werden.

Das Konzept wird von Forberg und Mayer [2002] wieder aufgegriffen und um die Erkennung von konkaven und konvexen Gebäudeteilen ausgebaut. Diese sind für die Realisierung komplexer Operatoren im Krümmungsraum notwendig. Um die Einschränkung auf rechtwinklige Strukturen zu umgehen, findet in [Forberg und Mayer 2003] eine Orthogonalisierung von geneigten Dachflächen und Wänden statt (vgl. Abbildung 3-15). Vor der eigentlichen Vereinfachung wird das Eingabemodell in ein Blockgebäude mit Flachdach überführt. Erst die rechtwinklige Struktur erlaubt die Anwendung der vorgestellten Operatoren.

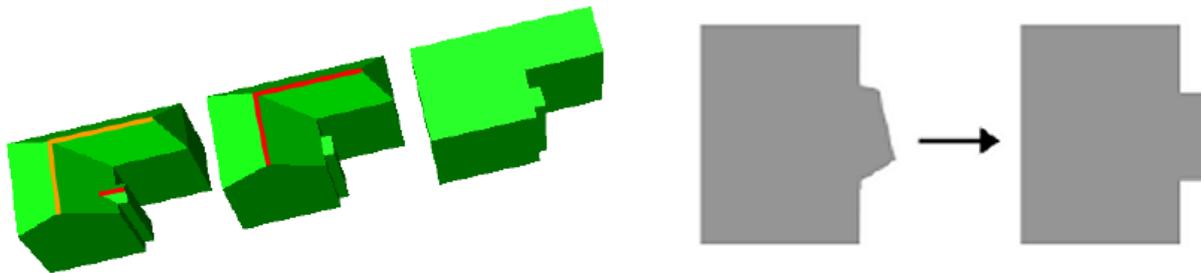


Abbildung 3-15: Orthogonalisierung von Dach- und Grundrissstrukturen (aus [Forberg 2005]).

Da die sequentielle Kombination der verwendeten Operatoren nicht immer zu den gewünschten Ergebnissen führte, stellt Forberg [2004] einen neuen Ansatz für einen Operator vor: Vorsprünge, Box- und Stufenstrukturen werden in einem einzigen Arbeitsschritt entfernt, indem parallele Flächen, die unter einer gewissen Distanz zueinander liegen, zueinander hinbewegt werden (siehe Abbildung 3-16).

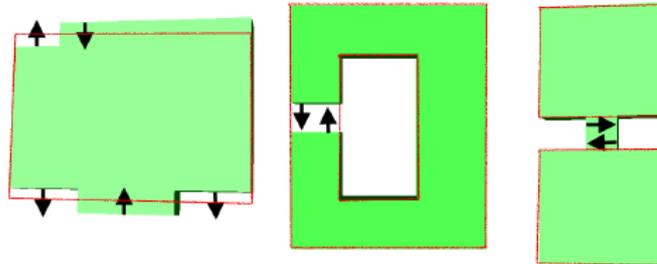


Abbildung 3-16: Zur Vereinfachung werden parallele Flächen zueinander bewegt (aus [Forberg 2005]).

Ein Verfahren, welches auf der Flächensimplifizierung aufbaut, wird in [Coors 2001] vorgestellt. Der Algorithmus von Garland und Heckbert [1997] wird erweitert, um visuell bedeutende Merkmale von Gebäuden beizubehalten. Der für die Wahrnehmung wichtige Glockenturm eines Gebäudes kann beispielsweise betont werden, indem das Gebäudemodell vereinfacht, die originale Detaillierung des Turms jedoch belassen wird (siehe Abbildung 3-17).

Erreicht wird dies mittels Dominanzwerten für Geometrieknoten. Die Vereinfachung einer Kante mit zwei Dominanzwerten erfährt durch das manuelle Einfügen einer zusätzlichen Ebenengleichung in die Fehlermetrik (siehe Abschnitt 3.3) einen Nachteil. Dadurch werden diese Kanten erst spät im Vereinfachungsprozess entfernt. Auf die Vereinfachung von Gebäudegeometrien unter Beibehaltung geometrischer Relationen wie rechtwinklige Wände wird in der Arbeit jedoch nicht eingegangen.

Für die Visualisierung von 3D-Stadtmodellen wird weiterhin eine Aggregation von Hintergrundgebäuden vorgeschlagen, um Landmarken, die sich im Blickfeld des Betrachters befinden, zu betonen. Durch die Hervorhebung von Schloss und Kirche in Abbildung 3-17 soll beispielsweise eine leichtere Orientierung im 3D-Stadtmodell von Darmstadt möglich sein. Realisiert wurde das Konzept mittels unterschiedlicher Detailstufen, die in einem hierarchischen R-Baum strukturiert sind. Die Aggregation von Gebäuden wird in den inneren Knoten verhindert, wenn deren Vereinigung die wichtigen Merkmale der enthaltenen Landmarken nicht

bewahren kann. Der Schwerpunkt der Arbeit liegt auf Echtzeitanwendungen, so dass kartographische Gesichtspunkte bei der Aggregation nicht berücksichtigt werden können.

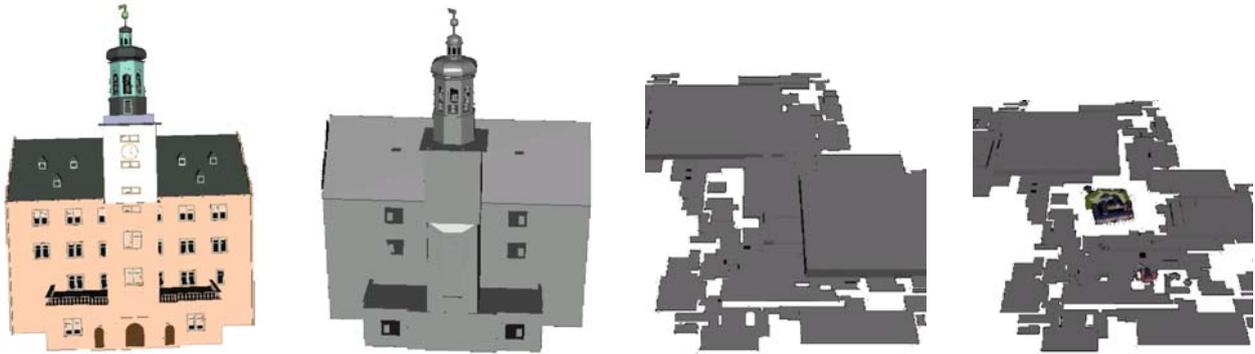


Abbildung 3-17: Der Glockenturm im Schloss von Darmstadt vor und nach der Generalisierung. Die Aggregation der Innenstadt von Darmstadt ohne und mit Betonung von Landmarken (aus [Coors 2001]).

Das Generalisierungsverfahren von Rau et al. [2006] basiert auf speziellen 3D-Gebäudemodellen, wie sie in [Rau und Chen 2003] beschrieben werden. Deren Rekonstruktion erfolgt aus einer Kombination von LIDAR- und Stereobildauswertung [Chen et al. 2004]. Sie bestehen aus einer Menge von disjunkten Polyedern, welche ausschließlich Flach- oder Pultdachform besitzen dürfen. Ein Sattel- oder Walmdach wird dabei aus zwei bzw. vier Polyedern geformt. In Abhängigkeit von der Entfernung des Betrachters, den Projektionsparametern und der Bildschirmauflösung wird für die 3D-Visualisierung eine Merkmalsauflösung bestimmt, ab der Gebäudeelemente nicht mehr vom Restgebäude differenzierbar sind. Die Vereinfachung des Gebäudemodells erfolgt in zwei Phasen. Zuerst werden Polyeder mit Pultdach betrachtet. Ist der Höhenunterschied des schrägen Dachpolygons geringer als die Merkmalsauflösung, so wird das Pultdach in ein Flachdach umgewandelt. Anschließend werden die Höhenunterschiede zwischen benachbarten Flachdächern untersucht. Bei einem Höhenunterschied kleiner der Merkmalsauflösung werden die Polyeder vereinigt. Der resultierende Polyeder erhält einen anteilmäßig an den Grundflächen gemittelten Höhenwert. Nach der Dachstruktur erfolgt die Vereinfachung des Gebäudegrundrisses. Hierzu werden schmale Wandelemente gesucht, die unter der Merkmalsauflösung liegen. Es wird unterschieden in innere Wände, die an einen anderen Polyeder angrenzen und in Wände der Außenhülle. Aufgrund der speziellen Struktur der Gebäudemodelle besitzen Wandpolygone genau vier Kanten und damit vier Nachbarpolygone. Zwei der Nachbarpolygone sind Teil des Grundrisses bzw. der Dachstruktur, so dass pro Wandpolygon nur zwei Nachbarwände betrachtet werden müssen. In Abhängigkeit von der vorhergehenden und nachfolgenden Wand wird das zu kleine Wandelement über einen „Wandkollapsoperator“ eliminiert. Für eine Sequenz von inneren Wänden, die alle an denselben Nachbarn grenzen, wird die obere und untere Kante in den Mittelpunkt der Kante kollabiert (siehe Abbildung 3-18). Grenzen die vorhergehenden und nachfolgenden Wände an unterschiedliche Polyeder, so wird der neue Punkt zur Kontaktstelle der zwei unterschiedlichen Nachbarn hinbewegt. Sind durch die Kollapsoperation nur Außenwände betroffen, so können deren Knoten entlang der Gebäudehauptachsen bis zu einer maximalen Distanz verschoben werden, um die Grundrissstruktur weiter zu vereinfachen.

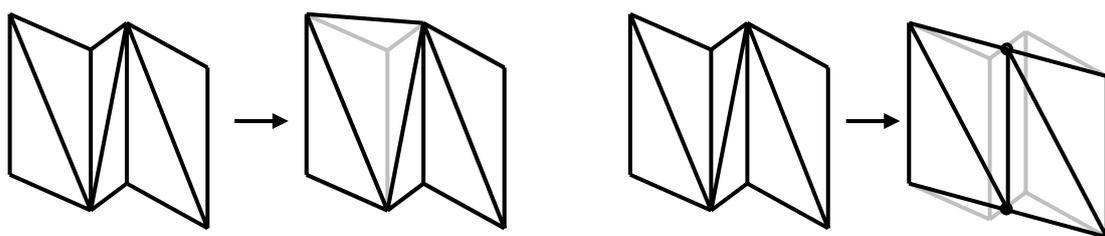


Abbildung 3-18: Ungültige (links) und gültige (rechts) Wandkollapsoperation. (nach [Rau et al. 2006]).

Ähnlich den Verfahren, die mittels Mustererkennung 2D-Grundrisspolygone generalisieren, stellt Lal [2005] ein Verfahren zur Erkennung von Gebäuden und Siedlungsstrukturen vor. Es wird unterschieden in einfache Gebäude mit unterschiedlichen Dachtypen wie Flach-, Pyramiden-, Giebel-, Walmdach etc. und komplexen Gebäuden mit kombinierten Dachtypen oder besonderen Grundrissformen. Die Strukturerkennung wird hierarchisch auf den Ebenen Einzelgebäude, Nachbarschaftsgebäude und Gebäudeblöcke durchgeführt. Dabei kommen Künstliche Neuronale Netze zum Einsatz, welche die Formparameter von Einzelgebäuden über den Grundriss und der Dachstruktur ableiten. Komplexe Gebäude werden dabei durch die Kombination einfacher Gebäude zusammengebaut. Aus den Einzelgebäuden werden dann mittels Perceptual Grouping Siedlungsstrukturen abgeleitet. Perceptual Grouping stammt aus dem Bereich Bildverstehen und basiert auf Erkenntnissen der Gestaltpsychologie. Diese entstand zu Beginn des 20. Jahrhunderts aufgrund der Beobachtungen von Ehrenfels [1890] und befasst sich mit der Wahrnehmung des Menschen. Das Ziel des Ansatzes ist die Gewinnung signifikanter Relationen aus Gebäuden und deren Anordnung zu bedeutsamen Siedlungsstrukturen. Die Gruppierung der Gebäude erfolgt dabei über die räumlichen Beziehungen Nachbarschaft, Kontinuität (Lage in einer Linie oder Kurve), Parallelität, Ähnlichkeit, Orientierung und Kollinearität. Wie die erkannten Gebäude und Siedlungsstrukturen zu generalisieren sind, ist jedoch nicht Thema der Arbeit.

Ein Problem, mit dem die meisten 3D-Generalisierungsverfahren zu kämpfen haben, ist die Vereinfachung der Dachstruktur. Dies ist darauf zurückzuführen, dass die verwendeten Operatoren nur mit Strukturen umgehen können, die aus rechtwinklig angeordneten Linien- und Flächensequenzen bestehen. Bevor der Grundriss generalisiert werden kann, formen diese Verfahren daher zunächst die Dachstruktur in ein Flachdach um. Kleine Grundriss- und Fassadendetails verschwinden also erst, nachdem alle groben Dachdetails eliminiert wurden. Darunter leidet jedoch merklich das charakteristische Erscheinungsbild von Gebäuden. Dies ist umso bedauerlicher, als dass es bei den vorherrschenden Gebäudemodellen keine detaillierten Fassadenstrukturen gibt. Somit ist die Vereinfachung der Dachform die wesentliche Unterscheidung der 3D-Generalisierung im Vergleich zur 2D-Grundrissgeneralisierung.

Die Verwendung von 3D-Mustern für Gebäude verspricht ein interessanter Ansatz für Gebäude mit Standardformen zu sein. Es wird sich jedoch zeigen müssen, wie viele verschiedene Muster notwendig sind, um alle erdenklichen Gebäudeformen zu unterstützen. Besonders für Gebäudekomplexe mit gemischten Dachformen und für prominente Gebäude wird es schwierig sein allgemeine Muster zu finden. Wenn allerdings die globale Gebäudeform bekannt ist, ergeben sich im Vergleich zu den iterativen lokalen Verfahren vielfältigere Möglichkeiten für die Generalisierung. Kapitel 5 zeigt beispielsweise, wie sich durch Typifizierung die Komplexität von Shed-, Parallel- und Grabendächern reduzieren lässt, indem die Anzahl der identischen Teildächer verringert wird. Die Charakteristik der Dachform bleibt dabei vollständig erhalten.

Ähnliches gilt für die Generalisierung von Fassaden, bei der die Fenster, Türen, Balkone, etc. unter Ausnutzung sämtlicher elementarer Vorgänge in der Anzahl reduziert und neu angeordnet werden. Dabei stellt sich die Generierung semantisch strukturierter Fassaden aus uninterpretierten Randbeschreibungen von Gebäuden als äußerst schwierig dar. Vielmehr müsste eine solche semantische Strukturierung aus der Rekonstruktion der Gebäudefassade stammen, wie sie beispielsweise in Kapitel 6 vorgestellt wird.

3.3 Flächensimplifizierung

Der Begriff Flächensimplifizierung (Surface Simplification) bezeichnet die Vereinfachung von polygonalen Flächenmodellen zum Zweck der effizienten Speicherung, Übertragung, Berechnung und Darstellung. Es existiert mittlerweile eine Reihe von Veröffentlichungen, welche die bekannten Algorithmen zusammenfassen (siehe beispielsweise [Heckbert und Garland 1997], [Krus et al. 1997], [Puppo und Scopigno 1997] und [Luebke et al. 2002]). Es wird unterschieden in Algorithmen für Höhenfelder, mannigfaltige und nicht-mannigfaltige Flächen (siehe z.B. [Garland 1999]). Nicht-mannigfaltige Flächen erlauben, dass sich drei oder mehr Polygone eine Kante teilen. Da 3D-Gebäudemodelle in der Regel als mannigfaltige Oberflächenmodelle erfasst werden, sollen hier die wichtigsten Algorithmen insbesondere aus dieser Klasse ausgeführt werden. Einige der Vereinfachungsoperatoren finden, wie in Kapitel 4 und 5 beschrieben, auch in den vorgestellten Generalisierungsverfahren für 3D-Gebäudemodelle ihre Anwendung. Die Generalisierung in Kapitel 5 vereinfacht Gebäudemodelle beispielsweise über Operatoren zur Erkennung und Entfernung von Merkma-

len (Abschnitt 3.3.9), welche mit dem Edge Collapse Operator (Abschnitt 3.3.3) realisiert sind und damit die Vorteile einer progressiven Speicherung aufweisen (Abschnitt 3.3.4).

3.3.1 Vertex Clustering

Eine sehr allgemeine Methode zur Simplifizierung von 3D-Dreiecksnetzen ist das Vertex Clustering. Dabei wird die Anzahl der Primitive im Objekt reduziert, indem mehrere Eckpunkte auf eine gemeinsame Position zusammengezogen werden (Cell Collapse Operator). Rossignac und Borrelt [1993] teilen dazu die Hüllgeometrie des zu vereinfachenden Objekts in regelmäßige, dreidimensionale Gitterboxen. Überdies erhält jeder Eckpunkt eine Gewichtung, die sich aus der Größe und Krümmung der angrenzenden Dreiecksflächen berechnet. Zur Simplifizierung der Geometrie werden alle Eckpunkte, die sich innerhalb derselben Zelle befinden, im Punkt mit der höchsten Gewichtung vereinigt. Anschließend werden alle Dreiecke aus dem Datensatz entfernt, bei denen zwei oder drei Ecken auf dieselbe Punktposition fallen.

Bei der Darstellung des vereinfachten Objekts können diese degenerierten Dreiecke zusätzlich als Punkt- oder Linienprimitive mit der übrigen Geometrie gerendert werden. Dadurch wird eine schematisch getreue Abbildung erreicht, da lange dünne Objektteile zumindest noch als Linien dargestellt werden. Low und Tan [1997] speichern bei der Vereinfachung der Vertex Cluster zusätzlich noch den Durchmesser um die Dicke der Linienprimitive entsprechend zu variieren. Durch Schattierung mittels dynamischer Normalenvektoren, erhalten die Linien auch noch ein zylinderförmiges Aussehen, wodurch sich das Erscheinungsbild des Objekts mit geringem Aufwand weiter verbessert.

Um auch die Qualität der vereinfachten Geometrie weiter zu optimieren, ersetzen Low und Tan [1997] das regelmäßige Gitter durch eine würfel- oder kugelförmige Zelle mit benutzerdefinierter Größe. Vor der eigentlichen Simplifizierung werden zunächst alle Eckpunkte des Dreiecksnetzes nach ihrer Gewichtung sortiert. Dann wird iterativ die Zelle um den Eckpunkt mit der höchsten Gewichtung gelegt und alle Punkte innerhalb der Zelle im Mittelpunkt vereinigt. Durch die Sortierung der Eckpunkte verschlechtert sich die Zeitkomplexität des Verfahrens jedoch von $O(n)$ auf $O(n \log n)$.

Die Verwendung eines regelmäßigen Gitternetzes ist nicht nur äußerst effizient bezüglich der Rechenzeit, sondern reduziert auch die Größe des Arbeitsspeichers, den der Algorithmus benötigt. Diese Vorteile nutzt Lindstrom [2000] für eine Out-of-Core Implementierung, um selbst große Datensätze, die nicht mehr vollständig in den Hauptspeicher passen, verarbeiten zu können. Der Algorithmus benötigt nur einen einzigen Durchlauf über die Eingangsdaten um ein simplifiziertes Modell zu erstellen. Der Speicherbedarf ist dabei unabhängig von der Größe des Originalmodells. Anders als beim ursprünglichen Vertex Clustering Ansatz werden die im Modell verbleibenden Eckpunkte jedoch nicht einfach auf die Position des am höchsten gewichteten Eckpunkts verschoben. Stattdessen wird die Position mittels der Fehlermetrik mit Quadriken (siehe [Garland und Heckbert 1997]) optimiert. Dazu wird für jeden Eckpunkt, der sich innerhalb der aktuellen Rasterzelle befindet, eine Quadrik aus allen angrenzenden Dreiecke aufsummiert. Dreiecke, die durch die Vereinfachung degenerieren, d.h. deren Eckpunkte sich nicht in drei verschiedenen Gitterzellen befinden, werden dabei ignoriert. Durch die Lösung eines Systems von linearen Gleichungen wird aus der Fehlermetrik die geometrisch optimale Position des Eckpunkts berechnet.

Schaefer und Warren [2003] verwenden einen dynamischen Octree, um Eckpunkte in beliebiger Auflösung aufzuteilen. Dadurch können Details im vereinfachten Modell exakter wiedergegeben werden. Durch die Baumstruktur liefert der Algorithmus ein vergleichbares Ergebnis wie Vertex Clustering Methoden mit theoretisch unlimitiertem Speicher.

Vertex Clustering Verfahren sind einfach, arbeiten sehr schnell und sind zudem äußerst robust gegenüber degenerierten Dreiecksnetzen. Die Qualität der resultierenden Modelle kann über die Größe der verwendeten Gitterbox gesteuert werden. Ein grobes Gitter hat eine aggressivere Dreiecksreduktion zur Folge wie ein feines. Vor allem bei groben Gittern werden Details jedoch nicht sehr gut bewahrt. Hinzu kommt, dass ein maximaler geometrischer Fehler nicht garantiert werden kann. Ferner ist die Simplifizierung des Modells auf eine vorher festgelegte Anzahl von Dreiecken nur durch Probieren möglich. Geometrische oder sogar topologische Garantien können durch die Algorithmen nicht gewährleistet werden. Hierzu müssen die endgültigen Positionen der Punkte eingeschränkt werden, wie das in Kapitel 5 getan wird. Die dort eingeführten Restriktionen bevorzugen Punktpositionen, die sich auf bestehenden Kanten und Flächen befinden, wodurch die Regelmäßigkeit der Gebäudeflächen bewahrt wird.

3.3.2 Vertex Decimation

Ein weiterer früher Algorithmus zur Flächensimplifizierung ist das Vertex Decimation Verfahren, welches in seiner ursprünglichen Form von Schröder et al. [1992] veröffentlicht wurde. Ein Dreiecksnetz wird durch wiederholte Anwendung des Vertex Removal Operators vereinfacht, der sowohl einen Knoten als auch die angrenzenden Kanten und Dreiecke entfernt (siehe Abbildung 3-19). Das durch diesen Vorgang entstandene Loch wird anschließend mit weniger Dreiecken wieder geschlossen. Klein und Kramer [1997] zeigen, dass die Triangulierung auf unterschiedlicher Art und Weise gelöst werden kann, wovon eine äquivalent der Half-Edge Collapse Operation ist.

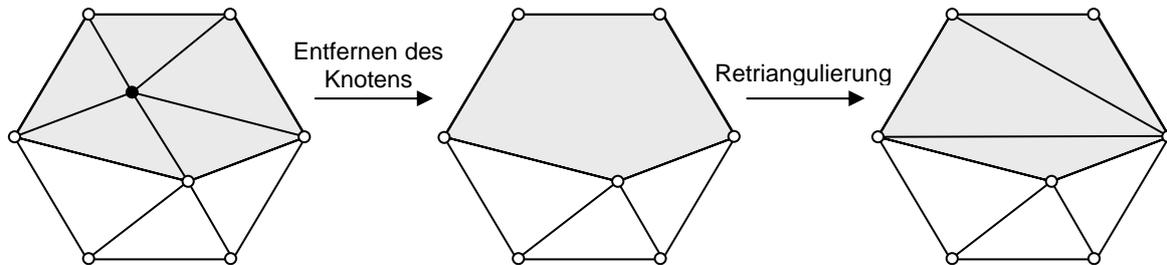


Abbildung 3-19: Beim Vertex Removal Operator wird ein Eckpunkt ausgewählt und mitsamt aller angrenzenden Dreiecke entfernt. Das entstandene Loch wird anschließend neu trianguliert (nach [Luebke et al. 2002]).

Vor der eigentlichen Simplifizierung teilt der Algorithmus die Eckpunkte zunächst in fünf Klassen ein. Die einfachen Eckpunkte sind lokal mannigfaltig und durch einen vollständigen Ring von Dreiecken umgeben. Ist dieser Ring nicht vollständig, so handelt es sich um einen Randpunkt. Alle anderen Punkte werden als nicht-mannigfaltig klassifiziert und für eine Entfernung nicht weiter in Erwägung gezogen. Die einfachen Eckpunkte werden aus den Winkeln und Attributen (z.B. Farbe oder Texturkoordinaten) der umgebenden Dreiecke weiter in innere Kanten und Ecken unterschieden.

Die Simplifizierung des Dreiecksnetzes erfolgt anschließend in mehreren Durchläufen. In jedem Durchlauf wird für alle Eckpunkte ein geometrisches Fehlermaß berechnet. Für einfache Punkte wird eine Durchschnittebene aus den umgebenden Dreiecken gebildet. Dazu gehen die Dreiecke nach deren Flächeninhalt gewichtet in die Berechnung der Ebenenparameter ein. Für Randpunkte und innere Kanten wird eine Durchschnitlinie aus den zwei Eckpunkten der (Rand-) Kanten gebildet. Das Fehlermaß berechnet sich dann als der Abstand des Punktes zu dessen Ebene bzw. Linie. Knoten, deren Fehlermaß unter einem Schwellwert liegen, werden aus dem Dreiecksnetz entfernt. Durch rekursives Aufspalten des Ringpolygons in Dreiecke wird das entstandene Loch wieder geschlossen. Da die als Ecken klassifizierten Punkte für das Erscheinungsbild als besonders wichtig erachtet werden, findet eine Löschung dieser Punkte nur in Ausnahmen statt. Nach jedem Durchlauf des Algorithmus wird der Schwellwert erhöht bis das resultierende Dreiecksnetz die gewünschte Detailstufe erreicht hat.

Da der Algorithmus in einem Durchlauf für alle Eckpunkte das Fehlerkriterium berechnet, ist dessen Zeitkomplexität linear abhängig von der Anzahl der Punkte. Er eignet sich daher insbesondere sehr gut, um überflüssige koplanare Geometrien effizient in einem einzigen Durchlauf zu entfernen. Die beschriebene Fehlermetrik führt jedoch zu weniger genauen Resultaten als präzisere Metriken. Auch ist der Grad der Dreiecksreduktion vor einem Durchlauf nicht bekannt, so dass eine gezielte Dreieckszahl des Ergebnismodells nur schwer realisierbar ist. Schroeder [1997] erweitert daher den Algorithmus, indem die Eckpunkte in einer Prioritätswarteschlange verwaltet werden. Die Simplifizierung kann dadurch schrittweise in der Reihenfolge des geringsten einzuführenden Fehlers erfolgen. Zur Dreiecksreduktion verwendet der Algorithmus den Halfedge Collapse Operator, um Punkte in einem Schritt zu entfernen und gleichzeitig die entstandenen Löcher wieder zu füllen. Ist keine weitere Dreiecksreduktion mehr möglich, so erlaubt das Verfahren die Aufteilung von Kanten und nicht-mannigfaltigen Eckpunkten mittels der Vertex Split Operation. Es entsteht dadurch zwar jeweils ein zusätzlicher Eckpunkt, aber macht eine weitere Vereinfachung möglich.

3.3.3 Edge Collapse

Eine weitere äußerst populäre Klasse von Algorithmen verwenden den so genannten (Full-) Edge Collapse Operator um Dreiecksnetze iterativ zu vereinfachen (siehe Abbildung 3-20). Dieser Operator wird beispielsweise im Generalisierungsverfahren von Kapitel 4 verwendet um Gebäudemerkmale zu entfernen. Zum ersten Mal wurde der Edge Collapse Operator von Hoppe et al. [1993] zur Flächensimplifizierung vorgeschlagen. Im Gegensatz zum Halfedge Collapse Operator, bei dem die beiden Endpunkte der Kante auf einen der beiden Punkte zusammengezogen werden, wird beim Edge Collapse Operator für den verbleibenden Eckpunkt eine neue Position berechnet. Die an die Kante angrenzenden Dreiecke degenerieren bei diesem Vorgang und werden anschließend aus dem Dreiecksnetz entfernt. Die Topologie des Dreiecksnetzes kann sich dabei durch das Schließen von Löchern in der Oberfläche ändern. Beim Vertex Pair Collapse (siehe [Garland und Heckbert 1997]) brauchen die beiden Punkte nicht mal mehr durch eine Kante verbunden sein, sondern es werden beliebige Knotenpaare zusammengezogen. Der Abstand der Knoten muss nur unter einem definierten Schwellwert liegen. Dadurch wird zusätzlich zur Geometrie auch die Topologie vereinfacht, indem separate Teile der Fläche zusammengefügt werden.

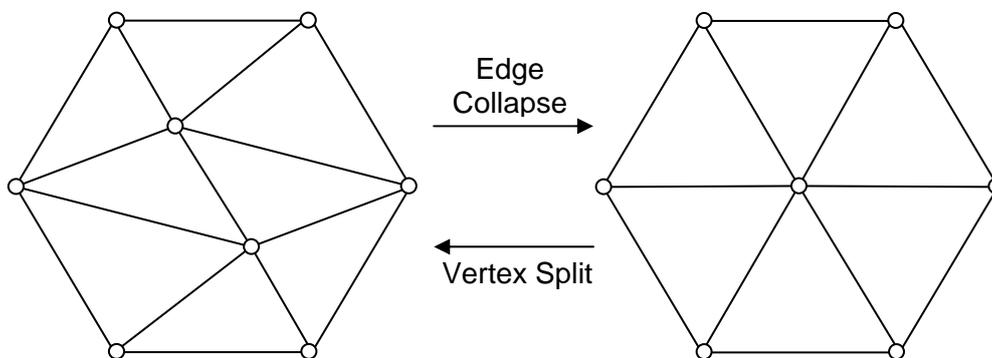


Abbildung 3-20: Edge Collapse Operator und Umkehrung Vertex Split (nach [Luebke et al. 2002]).

Die Vereinfachung des Flächenmodells wird wiederum über eine Fehlermetrik gesteuert. Diese bestimmt, welches Knotenpaar als nächstes zusammengezogen werden soll oder ob das Modell schon eine vorgegebene Mindestauflösung erreicht hat. Der Ansatz von Hoppe et al. [1993] minimiert hierfür eine Energiefunktion. Diese setzt sich zusammen aus drei Termen: die Einpassung des Dreiecksnetzes an eine vorgegebene Punktemenge, die Anzahl der Eckpunkte und ein Stellfederkriterium, welches die Optimierung auf ein lokales Minimum regelt. Wird beispielsweise ein Eckpunkt aus dem Dreiecksnetz entfernt, so verschlechtert sich zwar die Einpassung, der Term für die Anzahl der Eckpunkte nimmt jedoch zu. Das Knotenpaar welches die höchste Energiezunahme bzw. den kleinsten Energieverlust verursacht, wird in der nächsten Iteration zusammengezogen.

Der Algorithmus von Ronfard und Rossignac [1996] weist jedem Eckpunkt anfänglich die Menge von Ebenen der angrenzenden Flächen zu. Das Fehlermaß eines Eckpunktes ist das Maximum der quadratischen Entfernungen zu diesen Ebenen. Während der Vereinfachung werden dann in jedem Schritt die Mengen der zwei kontrahierten Eckpunkte vereinigt. Eine ähnliche Fehlermetrik wird in [Garland und Heckbert 1997] beschrieben. Jedem Knoten wird eine 4x4 Matrix zugewiesen, welche die Summe der quadrierten Abstände zu den Ebenen der angrenzenden Dreiecke repräsentiert. Aus diesen Quadriken leitet sich auch der englische Name der Fehlermetrik ab: Quadric Error Metrics. Der eingeführte Fehler der Kontraktion zweier Knoten ist die Summe der beiden Quadriken. Der Algorithmus sortiert alle möglichen Knotenpaare gemäß diesem Fehlermaß in eine Prioritätswarteschlange und zieht iterativ immer das Paar mit dem geringsten Fehler zusammen. Der neue Eckpunkt wird optimal im Zentrum der konzentrischen ellipsenförmigen Isofläche der Quadrik platziert. Dazu ist eine Invertierung der 4x4 Matrix notwendig, um das System linearer Gleichungen zu lösen. Nach jeder Iteration werden die Fehlerwerte der sich veränderten Knotenpaare aktualisiert und der Algorithmus fährt mit dem nächsten Knotenpaar fort.

Die Fehlermetrik mit Quadriken erlaubt auf effiziente und einfache Weise die Simplifizierung zu steuern. Da die Matrix symmetrisch ist, werden nur zehn Fließkommawerte benötigt, um die geometrische Abweichung zu repräsentieren. Daher hat das Verfahren zudem auch einen vergleichsweise geringen Speicherbedarf.

Trotzdem ist die Wiedergabetreue der simplifizierten Modelle von sehr hoher Qualität. Garland und Heckbert [1998] und Hoppe [1999] beschreiben jeweils Erweiterungen des Verfahrens um zusätzliche Attribute wie Farbe und Textur mit der Fehlermetrik zu behandeln. Lindstrom und Turk [1999] stellen eine alternative Lösung zur Berechnung der neuen Knotenposition vor. Die Quadrik wird nicht für jeden Eckpunkt gespeichert, sondern die Simplifizierung basiert auf einem inkrementellen Fehler. Dennoch werden bei diesem Simplifizierungsansatz qualitativ gute Resultate erzeugt.

3.3.4 Progressive Netze

Hoppe [1996] erweitert die oben beschriebene Energiefunktion desselben Autors. Die wesentlichste Neuerung ist jedoch die Einführung einer sehr effizienten Datenstruktur, um Dreiecksnetze in mehreren Detailstufen zu repräsentieren. Diese als Progressives Netz bezeichnete Struktur erlaubt eine schrittweise Verfeinerung und Übertragung des 3D-Objekts. Das Netz besteht aus einem durch Edge Collapse vereinfachten Basisnetz und einer Sequenz von Knotenteilungsoperationen (Vertex Split Operation). Eine Knotenteilung ist die Umkehroperation eines Edge Collapse. Sie ersetzt einen Eckpunkt durch zwei mit einer Kante verbundenen Knoten, wodurch zwei zusätzliche Dreiecke entstehen. Während der Simplifizierung zum Basisnetz wird die Sequenz von Knotenteilungsoperatoren in umgekehrter Reihenfolge zu den ausgeführten Edge Collapse Operationen gespeichert. Sie kodiert also ein Kontinuum von Simplifizierungsoperationen von dem Basisnetz zurück zum Originalmodell. Durch deren Ausführung kann das Ausgangsobjekt sowie jede beliebige Zwischenrepräsentation erzeugt werden. Jede Zwischenrepräsentation kann auch in jede andere Zwischenrepräsentation gewandelt werden. Da die Edge Collapse und Knotenteilungsoperationen schnell genug in Echtzeit ausgeführt werden können, sind dynamische und blickwinkelabhängige Simplifizierungen möglich. Sander et al. [2001] zeigt, wie Progressive Netze um Texturen erweitert werden können, ohne dass sich Bildbereiche durch die Simplifizierung überdehnen. Da die Generalisierung aus Kapitel 4 auf dem Edge Collapse Operator basiert, können auch Generalisierungsfolgen als Progressives Netz gespeichert werden, um dynamisch entsprechende Detailstufen zu erzeugen.

3.3.5 Voxel-Basierte Simplifizierung

Ein Ansatz mittels Verfahren der Signalverarbeitung ist die voxel-basierende Simplifizierung von He et al. [1995]. Es wird nicht nur die Geometrie, sondern auch die Topologie bzw. die Anzahl der Löcher, auch Genus genannt, des Objekts vereinfacht. Im ersten Schritt wird über ein Abtastverfahren (Volume Sampling) eine volumetrische Repräsentation des Modells erstellt. Der polygonalen Geometrie wird dann ein dreidimensionales Voxelgitter überlagert. Voxel innerhalb der Geometrie erhalten den Wert Eins, alle anderen den Wert Null. Dabei können nicht nur Einzelobjekte sondern auch mehrere Objekte verarbeitet werden. Als nächstes werden kleine Objektmerkmale von hoher Frequenz mittels eines Tiefpassfilters schrittweise eliminiert. Das Ergebnis ist ein volumetrisches Modell mit geringerer Auflösung. Aus diesem wird mit Hilfe des Marching-Cubes-Algorithmus (siehe [Lorenson und Cline 1987]) wieder eine Dreiecksfläche erzeugt. Über die Voxelgröße kann die gewünschte Detailstufe eingestellt werden. Leider werden hochfrequente Details wie scharfe Kanten oder Ecken durch den Algorithmus nicht gut erhalten. Er eignet sich also weder für technische CAD Modelle noch für Objekte, deren Erscheinungsbild sehr stark von diesen Merkmalen abhängen. Die voxelbasierten Ansätze sind damit ungeeignet für Generalisierungsaufgaben.

3.3.6 Bild-Basierte Simplifizierung

Die oben beschriebenen Verfahren verwenden geometrische Fehlermetriken um die Simplifizierung zu bewerten und zu steuern. Ein andersartiger Ansatz wird in [Lindstrom und Turk 2000] vorgestellt. Das Ergebnis der Simplifizierung wird mit Hilfe von gerenderten Bildern verifiziert. Dadurch wird ein guter Kompromiss zwischen Attributen, die das Erscheinungsbild beschreiben (z.B. Normalen, Farbe, Textur), und geometrischer Verformung erreicht. Der Algorithmus gehört zwar zu den langsamsten, bewahrt aber sehr gut die Silhouette des Objekts, vereinfacht drastisch nicht sichtbare Teile des Modells und erlaubt eine textursensitive Simplifizierung.

Als Simplifizierungsoperator kommt wieder der Edge Collapse Operator zum Einsatz und die Punktkandidaten werden in einer Prioritätswarteschlange nach deren geschätzten, einführenden Fehler sortiert. Die Fehler-

abschätzung geschieht über den Vergleich des gerenderten Modells vor und nach der Simplifizierung aus verschiedenen Blickwinkeln. Dabei kommt eine Technik zum Einsatz, welche durch geschickte Ausnutzung der Grafikhardware sehr schnell die Bilder für die Berechnung der Fehlerkosten aktualisiert. Da bei dem Ansatz das zu vereinfachende Objekt sehr häufig und in jeder Vereinfachungsstufe gerendert werden muss, eignet es sich eher für mittelgroße Modelle. Die Autoren verwenden daher einen geometrischen Simplifizierungsansatz [Lindstrom und Turk 1998], um das Modell zunächst auf eine angemessene Komplexität zu vereinfachen. Bei der Generalisierung von Gebäuden ist jedoch durchaus erwünscht, dass sich das Aussehen von Objekten verändert, sofern deren charakteristische Merkmale erhalten bleiben.

3.3.7 Simplifizierung mittels Alpha Shapes

Die so genannten Alpha Shapes fanden ursprünglich ihre Anwendung in der Rekonstruktion von Flächen aus Punktwolken oder der Darstellung von Molekülen. Bei diesen Verfahren befindet sich eine gegebene Punktmenge in einem mit Material gefüllten Raum. Eine Kugel mit Radius α rollt über diese Punkte und entfernt aus dem erreichbaren Raum das Material. Dabei darf die Kugel allerdings keinen Punkt überlappen und wird dadurch am Vordringen in alle Bereiche gehindert. Das verbleibende Material besitzt eine Oberfläche bestehend aus Kugelflächen, Kreissegmenten und einer Teilmenge der vorgegebenen Punkte. Die Kugelflächen werden anschließend durch Dreiecke und die Kreissegmente durch gerade Kanten ersetzt um den Alpha Shape zu erhalten.

El-Sana und Varshney [1998] verallgemeinern die Alpha Shapes auf polygonale Modelle. Die Simplifizierung eines polygonalen Objekts erfolgt, indem wieder eine Kugel mit Radius α darüber rollt und alle Regionen geschlossen werden die durch die Kugel nicht erreichbar sind. Bajaj und Kim [1988] haben schon früher gezeigt, dass die Bewegung einer Kugel über eine Dreiecksfläche gleichbedeutend mit der Bewegung eines Punkts auf einer Dreiecksfläche ist, die um den Radius der Kugel gewachsen ist. Das Schließen von Löchern kann also durch ein α -Wachstum von Dreiecken und anschließender Vereinigungsoperation realisiert werden. Mittels heuristischer Regeln zur Erkennung von Löchern können diese auch selektiv geschlossen werden. Durch Vergrößerung des Kugelradius schließen sich allmählich alle Konkavitäten des Objekts.

3.3.8 Erscheinungsbild bewahrende Simplifizierung

Algorithmen, die nicht allein den geometrischen Fehler der modifizierten Eckpunktpositionen betrachten, sondern auch auf das Erscheinungsbild achten, sind beispielsweise in [Cohen et al. 1998] und [Jang et al. 2003] beschrieben. Vor der Vereinfachung werden Farb- und Krümmungsinformationen in Texturdaten umgewandelt und entsprechende zusätzliche Texturfehlermetriken eingeführt. Dadurch lassen sich die Veränderungen in der Bildschirmdarstellung messen und den Verlauf der Vereinfachung steuern. Um bedeutende Merkmale bei der Vereinfachung so lange wie möglich zu bewahren, lassen Cignoni et al. [1998], Kho und Garland [2003], Li und Watson [2001] und Pojar und Schmalstieg [2003] die entsprechenden Bereiche im Modell vom Operateur manuell markieren. Die Vereinfachung verläuft noch immer dreiecksbasiert. Es wird nur die Reihenfolge der Operationen verändert. Das generelle Problem, die Merkmale zu detektieren und konsistent zu entfernen, wird nicht angesprochen. Um scharfe Kanten im Modell durch die Vereinfachung nicht zu verlieren, wurden ferner Methoden des Remeshing oder Resampling entwickelt [Botch und Kobbelt 2001], [Vorsatz et al. 2001], [Marinov und Kobbelt 2005], [Alliez et al. 2005].

3.3.9 Erkennung und Entfernung von Merkmalen

Eine Gruppe von Verfahren, die sehr speziell auf von Menschenhand geschaffene Objekte und Modelle ausgerichtet ist, sind die Algorithmen zur Erkennung und Entfernung von Merkmalen (Feature Detection and Removal). Einen Überblick über das Themengebiet gibt beispielsweise Wu und Liu [1996]. Ein Merkmal wird von Pratt und Wilson [1985] als eine „Region von Interesse auf der Oberfläche eines (Bau-) Teils“ definiert. Eine weitere, eher technische Definition gibt Ribelles et al. [2001]: „Ein Merkmal ist eine zusammenhängende Region auf einer Oberfläche, welche sich einfach vom Rest separieren lässt“. Die für die Generalisierung wichtigste Eigenschaft dieser Klasse von Algorithmen ist, dass Objektteile als ganzes bzw. als Gruppe von Dreiecken identifiziert und auf einmal konsistent aus dem Datensatz entfernt werden. Dadurch wird die Semantik des Objekts besser beachtet.

In der Arbeit von Ribelles et al. [2001] wird ein Algorithmus definiert, welcher als Eingabe das triangulierte Oberflächenmodell eines zusammenhängenden Polyeders akzeptiert. Der Algorithmus teilt das Originalmodell mit einer Reihe von Ebenen und klassifiziert die Dreiecke als innerhalb (IN), außerhalb (OUT) oder aufliegend (ON) hinsichtlich der Teilungsebene. Benachbarte und identisch klassifizierte Dreiecke werden in einem Cluster-Graphen gebündelt. Jeder Knoten im Graph, welcher keine aufliegenden Dreiecke besitzt, ist ein potentielles Merkmal. Da die Definition eines Merkmals oftmals applikationsspezifisch ist, wird von den Autoren nur eine generelle Gültigkeitsdefinition gegeben: die Entfernung eines Merkmals hat ein Loch im Basismodell und Merkmal zur Folge. Können die Löcher durch die Anknüpfung von koplanaren Dreiecken an die an die Löcher angrenzenden Flächen geschlossen werden, so ist das Merkmal gültig, andernfalls ungültig. Das Ergebnis sind triangulierte Oberflächenmodelle vom Hauptkörper und den entkoppelten Merkmalen und einer Menge boolescher Operatoren (Vereinigung oder Differenz) zu deren Kombination.

Weitere Verfahren teilen die Modelle in konvexe Teile auf [Kim 1992] [Woo 1982] oder verwenden Topologiegraphen, um anhand der Beziehungen zwischen Flächen und Kanten Merkmale zu erkennen [De Floriani 1989]. Dies kann zusätzlich mit geometrischen Analysen kombiniert werden [Gavankar und Henderson 1990], [Marefat und Kashyap 1990]. Lu et al. [1999] und Jang et al. [2005] untersuchen Geometrien auf Schlaufen, die durch Kanten gebildet werden. An diesen Schlaufen wird das Modell in zwei Teile aufgeteilt und rekursiv weiter untersucht. Dadurch entsteht ein hierarchischer Baum. Merkmale mit gleicher oder korrelierter geometrischer Wichtigkeit werden gleichzeitig und in einem Verarbeitungsschritt entfernt.

Zusammenfassend sind Verfahren zur Flächensimplifizierung allgemeiner Modelle nicht gut geeignet für die Generalisierung von 3D-Gebäudemodellen. Sie wurden für die Vereinfachung von sehr detaillierten Dreiecks- und Polygonnetzen entwickelt, welche aus Tausenden oder gar Millionen von Flächenprimitiven bestehen. Die eingesetzten Fehlermetriken garantieren eine möglichst geringe geometrische, topologische oder visuell darstellbare Abweichung vom Originalmodell. Dies ist für die Generalisierung für sich genommen jedoch noch nicht ausreichend oder sogar nicht mal erwünscht. Denn heute verfügbare 3D-Stadtmodelle bestehen aus sehr vielen Einzelgebäuden die allesamt nur eine vergleichsweise geringe Polygonzahl aufweisen. Üblich sind Modelle mit einer Komplexität von wenigen hundert bis tausend Dreiecken. Das bedeutet, dass die Netze für große Teile der Modelle keine Approximation von glatten Oberflächen sind, sondern es sich tatsächlich um Objekte mit planaren Flächen und scharfen Kanten handelt. Des Weiteren berücksichtigen die oben beschriebenen Fehlermetriken nicht die bei Gebäuden häufig vorkommenden geometrischen Relationen wie Parallelität und Rechtwinkligkeit von Fassadenflächen. Selbst eine geometrische Abweichung vom Original wird eher akzeptiert als die Missachtung der geometrischen Relationen. Die Verschiebung einer Fassadenwand um ein bis zwei Meter wird beispielsweise als weniger störend empfunden als wenn durch eine geringfügige Änderung in der Position eines Eckpunkts anschließend die Wand schräg im Raum zu liegen kommt. Oftmals sind aber auch die eingesetzten Simplifizierungsoperatoren nicht in der Lage, das Gebäudemodell noch geeignet zu vereinfachen. Das Schlepptdach des sehr einfach gehaltenen 3D-Gebäudemodells aus Abbildung 3-21 kann beispielsweise mittels des Edge Collapse Operators nicht weiter vereinfacht werden, ohne die Planarität der Wände zu verletzen und uneinheitliche Traufhöhen zu generieren. Vielmehr würde man von einer Gebäudegeneralisierung in diesem Fall die Erzeugung eines Modells mit einem symmetrischen Satteldach erwarten.

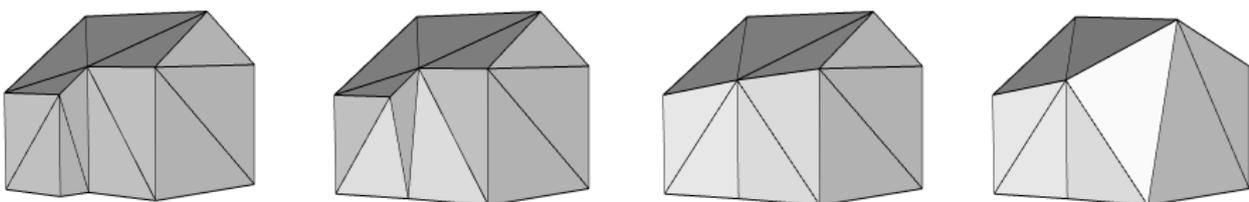


Abbildung 3-21: Vereinfachung eines 3D-Gebäudemodells mit Schlepptdach durch die QSlim Software ([Garland und Heckbert 1998]).

Weitere Probleme der Flächensimplifizierung sind, dass sich durch die Vereinfachung die Charakteristiken von Gebäudeteilen ändern können oder diese anschließend Charakteristiken enthalten, die vorher nicht da waren. Auch wenn das Ergebnis der Flächensimplifizierung den Ansprüchen einer Generalisierung genügt,

so ist nicht garantiert, dass dies auch auf die Zwischenergebnisse zutrifft. Dazu kommt, dass aufgrund des Fehlens von Qualitätskriterien für die 3D-Generalisierung nicht entscheidbar ist, welche Zwischenergebnisse überhaupt bedeutungsvoll sind. Wie in Kapitel 4 gezeigt wird, muss die Flächensimplifizierung also für die Generalisierung erweitert bzw. angepasst werden. Dies erfolgt, wie bei den Verfahren zur Merkmalerkennung und Entfernung, bei denen nach Gebäudeteilen mit spezifischen geometrischen Eigenschaften gesucht werden, die anschließend mit angepassten Operatoren zu entfernen sind.

4 3D-Generalisierung mittels Flächensimplifizierung

Nachdem in den vorangegangenen Kapiteln die Grundlagen zu 3D-Stadtmodellen, Generalisierung und Flächensimplifizierung erläutert wurden, soll nun das erste von zwei neuen Generalisierungsverfahren für 3D-Gebäudemodelle vorgestellt werden. Das Verfahren vereinfacht in einem automatischen Prozess 3D-Modelle von Gebäuden, die als Polygonnetze in Randbeschreibung gegeben sind. Zur Modellvereinfachung werden Operatoren aus dem Bereich der Flächensimplifizierung verwendet, um in Kombination Merkmale konsistent zu eliminieren. Ein Merkmal ist ein Gebäudeelement, welches vom Restgebäude entfernt werden kann, ohne dessen globales Erscheinungsbild zu zerstören. Bei der Ausführung wird besonders auf die korrekte Behandlung der geometrischen Relationen Koplanarität, Parallelität und Rechtwinkligkeit von Polygonflächen geachtet, um das typische Erscheinungsbild der Gebäude nicht zu zerstören. Während des Generalisierungsvorgangs wird daher dem geometrischen Fehler weniger Bedeutung beigemessen als der Bewahrung einer korrekten Gebäudestruktur. Da die Merkmale in einer räumlich begrenzten Umgebung betrachtet werden, handelt es sich hier um ein lokales Verfahren. Demgegenüber wird in Kapitel 6 ein globales Verfahren vorgestellt, bei dem der Untersuchungsbereich das Gesamtgebäude ist.

4.1 Die iterative Generalisierung

Die Generalisierung von 3D-Gebäudemodellen geschieht in dem vorgestellten Verfahren in zwei Schritten. Siehe hierzu auch Abbildung 4-1 für eine exemplarische Darstellung des Ansatzes in einer 2D-Ansicht. Im ersten Schritt wird die Geometrie des Gebäudemodells iterativ vereinfacht. Hierfür kommt ein Verfahren zur Erkennung und Entfernung von Merkmalen zum Einsatz, welches das Modell in jedem Iterationsschritt nach den unterstützten Merkmalen durchsucht, diese anhand der Signifikanz bewertet und das Merkmal mit dem geringsten Wert entfernt. Dies wird solange wiederholt, bis das Modell die gewünschte Komplexität erreicht hat, bzw. bis sich keine Kleinstelemente mehr aus dem Modell entfernen lassen. Durch die Vereinfachung kann sich das Gebäudemodell in der Lage und der Ausdehnung verändern. Werden beispielsweise nur Ausbuchtungen entfernt, so schrumpft das Modell leicht. Um diesen Effekt auszugleichen, wird im zweiten Schritt die vereinfachte Geometrie mittels einer Formeinpassung an die Originalgeometrie angeglichen. Dadurch weist das generalisierte Gebäudemodell die kleinstmögliche geometrische Abweichung vom Originalmodell auf.

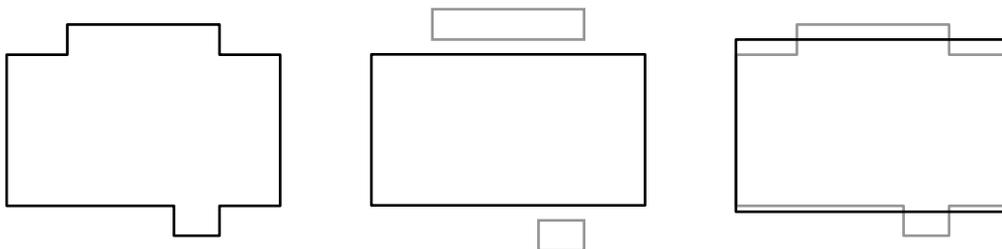


Abbildung 4-1: Gebäudegrundriss (links), vereinfachter Grundriss mit den entfernten Merkmalen (Mitte) und an die Originalgeometrie eingepasster, generalisierter Grundriss (rechts).

Die vorgestellte 3D-Generalisierung von Gebäuden ist eine Vereinfachung der entsprechenden Modelle unter Erhaltung der geometrischen Relationen Koplanarität, Parallelität und Rechtwinkligkeit der Gebäudeflächen. Da diese Relationen in der verwendeten Randbeschreibung nicht explizit gespeichert sind, wird eine zusätzliche Datenstruktur, der so genannte Relationsgraph, eingeführt. Dessen Struktur, Aufbau und Funktion wird im nächsten Abschnitt ausführlich beschrieben. Vor der Ausführung einer Vereinfachungsoperation kann mit Hilfe des Graphen die Auswirkungen auf die Relationen überprüft und bei Nichteinhaltung die Operation gegebenenfalls gestoppt werden. Auch eine Verzögerung ist möglich, da durch die Vereinfachung Flächen zu einem späteren Zeitpunkt wegfallen und somit die betroffenen Relationen ihre Gültigkeit verlieren kön-

nen. Weiterhin wird der Graph genutzt um nach der Vereinfachung die verbleibenden Relationen als Zwangsbedingungen in die Formeinpassung zu integrieren. Die Flächen dürfen bei diesem Vorgang nur in einer Art und Weise verändert werden, wie es die geometrischen Relationen des Graphen zulassen. Dadurch können beispielsweise Änderungen an den Gebäudeflächen auf parallele Bewegungen in Bezug zur ursprünglichen Lage beschränkt werden. Da der Relationsgraph das zentrale Element des Generalisierungsansatzes ist, soll dieser zunächst genauer erläutert werden.

4.2 Der Relationsgraph

Eine mittels Graphen realisierte Modellierung der geometrischen Beziehungen, die zwischen den Flächen eines oder mehrerer Objekte bestehen, wurde beispielsweise schon von Suh und Wozny [1997] verwendet.

Im Relationsgraph werden die drei geometrischen Relationen Koplanarität, Parallelität und Rechtwinkligkeit verwaltet. Die Knoten des Graphen repräsentieren dabei die polygonalen Flächen, welche durch Relationskanten miteinander verbunden sind. Abbildung 4-2 zeigt beispielhaft einen Fassadenausschnitt mit dem zugehörigen Relationsgraph. Die Flächen 1 und 5 liegen in einer Ebene und sind demnach mit einer Relation für Koplanarität (blau gestrichelt) im Graph verbunden. Mit diesen beiden Flächen ist Fläche 3 über eine Relation für Parallelität (grün) verknüpft. Es ist anzumerken, dass die zwei Flächen 2 und 4 auch als parallel betrachtet werden, obwohl deren Normalenvektoren in die entgegengesetzte Richtungen zeigen. Die restlichen (schwarzen) Relationen drücken eine Rechtwinkelbeziehung aus. Man sieht, dass in diesem Beispiel alle Flächen miteinander in irgendeiner geometrischen Beziehung stehen.

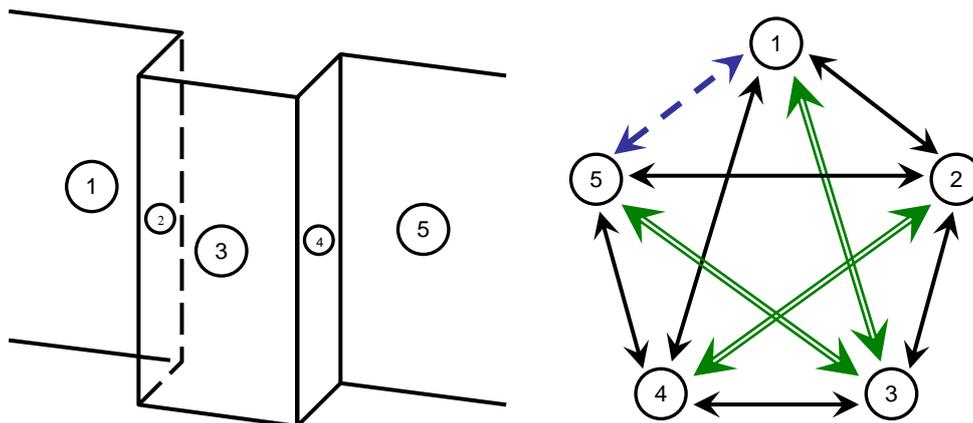


Abbildung 4-2: Fassadenausschnitt mit Ausbuchtung und der entsprechende Relationsgraph.

Der Relationsgraph beschreibt die Bedingungen, die im Modell erfüllt sein müssen. Würde eine Generalisierungsoperation eine Bedingung verletzen, so ist diese Operation ungültig und darf nicht ausgeführt werden. Sind zwei Flächen beispielsweise durch die geometrische Relationskante Rechtwinkligkeit miteinander verbunden, so muss der Winkel zwischen diesen beiden Flächen auch nach der Generalisierungsoperation zwingend 90° sein. Dasselbe gilt für koplanare Flächen. Eine Ausnahme bildet nur die Parallelität: Da die Koplanarität eine Sonderform der Parallelität darstellt, darf eine Parallelität im Relationsgraph nach der Generalisierung in eine Koplanarität umgewandelt werden. Dieser Fall tritt beispielsweise beim Entfernen von Ausbuchtungen auf (siehe Abbildung 4-3). Die Frontseite der Ausbuchtung wird durch die Generalisierungsoperation in dieselbe Ebene wie die vorher parallelen Fassadenflächen gezogen. Folglich können die Parallelitätskanten zu Koplanaritätskanten „hochgestuft“ werden. Für die wegfallenden Seitenflächen werden die Relationsknoten und alle indizierenden Kanten aus dem Graph gelöscht. Damit sind alle Bedingungen im Relationsgraph weiterhin erfüllt. Die Generalisierungsoperation ist gültig und darf somit durchgeführt werden.

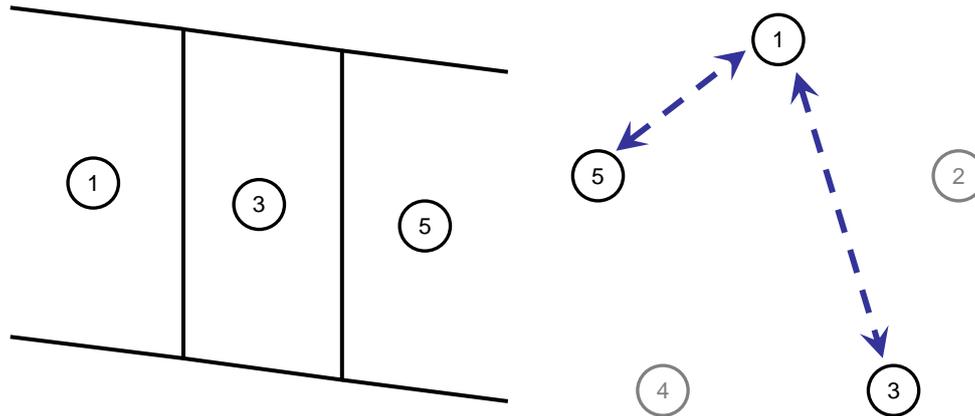


Abbildung 4-3: Generalisierter Fassadenausschnitt und der resultierende Relationsgraph.

Eine weitere Bedingung, die jedoch nicht explizit im Relationsgraph modelliert wird, ist die Planarität der Polygonflächen. Eine Operation ist nur dann gültig, wenn alle Polygonpunkte einer modifizierten Fläche auch nach der Generalisierung wieder in einer Ebene liegen. Wird beispielsweise ein Knoten eines Vierecks entlang des Normalenvektors der Fläche verschoben, so ist die Fläche nicht mehr planar und die Operation damit ungültig. Die Vereinfachungsoperatoren sollten zwar die Planarität von Flächen einhalten, trotzdem wird nach jeder Operation die von der Vereinfachung betroffenen Flächen auf Planarität hin überprüft und die Operation bei Nichteinhaltung wieder rückgängig gemacht.

Aufbau des Relationsgraph

Um den Relationsgraph erstellen zu können, werden explizite Informationen zu denen im 3D-Gebäudemodell befindlichen geometrischen Relationen benötigt. Diese sollten idealerweise aus der Modellierung oder Rekonstruktion des Modells stammen (siehe z.B. [Grün und Wang 2001]). Brenner [2004] beschreibt ein Verfahren, um Rechteckszerlegungen von 2D-Gebäudegrundrissen mittels so genannter „schwacher CSG Primitive“ zu modellieren. Die 2D-Primitive sind durch Zwangsbedingungen wie Parallelität, Rechtwinkligkeit, Punkt auf Linie etc. miteinander verknüpft und dienen als Ausgangsbasis für eine (semi-) automatische Rekonstruktion von Gebäuden aus Laserdaten. Diese Zwangsbedingungen könnten für das rekonstruierte 3D-Gebäudemodell direkt in den Relationsgraphen einfließen.

Sind die geometrischen Relationen nicht explizit modelliert, so müssen diese erst aus dem 3D-Gebäudemodell abgeleitet werden. Hierfür bietet sich die Ebenengleichung der Flächen in Hesse'sche Normalform ($E: \vec{n} \cdot \vec{x} = d$) an. Sie eignet sich sowohl für den Distanz- als auch für den Winkelvergleich. Der Winkel zwischen zwei Flächen berechnet sich als der Arkuskosinus des Punktprodukts der beiden Normalenvektoren. Im Prinzip kann hier auch mit Schwellwerten gearbeitet werden. Das funktioniert jedoch nur zuverlässig, solange der Erfassungsfehler kleiner als die minimal erfasste Distanz bzw. Winkel ist. Abbildung 4-4 zeigt einen Fall, wo der Fehler größer ist als der Schwellwert, der notwendig wäre, um die schräg liegenden Flächen, die in der Realität in einer Ebene liegen, auch als koplanar zu identifizieren.

In so einem Fall können die geometrischen Relationen nur durch einen menschlichen Operateur zweifelsfrei identifiziert werden. Vor der Weiterverarbeitung sind die gefundenen Relationen durch geometrische Umformung des Modells zu erzwingen. Ansonsten können die Relationen nicht mehr automatisch während des Generalisierungsprozess auf ihre Gültigkeit hin überprüft werden.

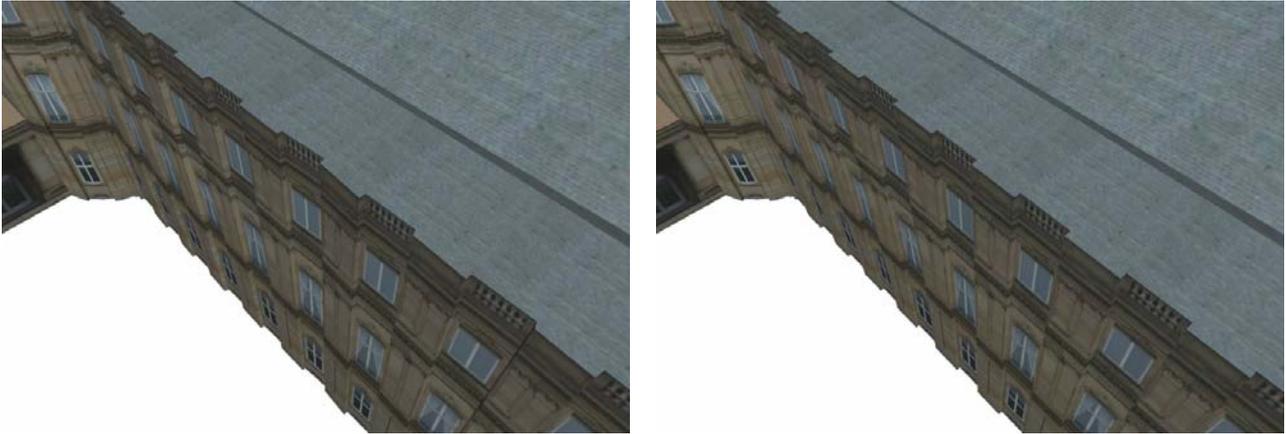


Abbildung 4-4: Die geometrischen Relationen des Gebäudemodells können aufgrund von Erfassungsungenauigkeiten nicht automatisch erkannt werden (links). Die Flächen des Gebäudemodells müssen erst von einem menschlichen Operateur manuell ausgerichtet werden (rechts).

Prüfung von Relationen

Die polygonalen Flächen der 3D-Gebäudemodelle müssen zwingend planar sein, so dass für jede Fläche eine eindeutige Ebenengleichung in Hesse'scher Normalform berechnet werden kann. Die geometrischen Relationen der Flächen lassen sich dann sehr effizient mittels des Normalenvektors \vec{n} und dem Abstand d auf deren Gültigkeit hin überprüfen. Hierbei kommt auch die Formel $\arccos(\vec{n}_1 \cdot \vec{n}_2)$ für die Berechnung des Schnittwinkels zweier Ebenen wie folgt zum Einsatz:

1. **Rechtwinkligkeit:** Der Kosinus von 90° ergibt den Wert null. Zwei Flächen sind daher rechtwinklig, wenn der Absolutbetrag des Kreuzprodukts der beiden Normalenvektoren kleiner ε ist ($|\vec{n}_1 \cdot \vec{n}_2| > \varepsilon$).
2. **Parallelität:** Der Kosinus von 0° und 180° ergibt die Werte minus eins und plus eins. Zwei Flächen sind parallel, wenn der Absolutbetrag des Kreuzprodukts der beiden Normalenvektoren größer $1 - \varepsilon$ ist ($|\vec{n}_1 \cdot \vec{n}_2| > 1 - \varepsilon$).
3. **Koplanarität:** Zwei Flächen sind koplanar, wenn die Bedingung für Parallelität erfüllt ist und der Absolutbetrag der Differenz der beiden Abstände d_1 und d_2 kleiner ε ist ($|d_1 - d_2| < \varepsilon$).

Optimierung

Für jede potentielle Vereinfachungsoperation müssen die geometrischen Relationen der sich dadurch verändernden Flächen auf ihre Gültigkeit hin überprüft werden. Des Weiteren gehen die Relationen zur endgültigen Formeinpassung der erzeugten Detaillierungsstufe als Bedingungen in einen Ausgleichungsprozess nach kleinsten Quadraten ein (Abschnitt 4.4). Je nach Komplexität des Ausgangsmodells und des eingesetzten Vereinfachungsoperators wirkt sich dies merklich auf die Leistung des Generalisierungsverfahrens aus. Daher sollte die Anzahl der effektiv genutzten Relationen möglichst gering sein.

Zu diesem Zweck werden die Kanten im Relationsgraph in Abhängigkeit von deren Rechenzeit unterschiedlich gewichtet. Das Gewicht gibt damit die Kosten an, die erbracht werden müssen, um die Relation zu prüfen und im Ausgleichungsprozess zu berücksichtigen. Die Rechtwinkligkeit bekommt dabei das höchste und die Koplanarität das niedrigste Gewicht. Der genaue Wert ist bei diesen drei Relationen nicht von Bedeutung, da es nicht möglich ist eine der drei Relationen durch Kombination der beiden anderen Relationen zu ersetzen. Dies kann sich jedoch durch Hinzunahme weiterer Relationstypen ändern. Durch die Einschränkung auf die drei Relationen ergibt sich die folgende Gewichtungstabelle (siehe Tabelle 4-1), welche die Größe des sich aus dem gewichteten Graphen resultierenden Gleichungssystems und damit die relative Rechenzeit für den Ausgleichungsprozess widerspiegelt:

Geometrische Relation	Gewichtung
Koplanarität	1
Parallelität	2
Rechtwinkligkeit	3

Tabelle 4-1 Kostentabelle für geometrische Relationen im Relationsgraphen.

Für den nun gewichteten Graphen werden die Kanten ermittelt, die alle Knoten im Graphen mit minimalen Gesamtkosten verbinden. Dies ist auch als das Minimalgerüst (Minimum Spanning Tree) bekannt und lässt sich mit dem Algorithmus von Kruskal [1956] oder Prim [1957] berechnen. Der Relationsgraph des Fassadenausschnitts aus Abbildung 4-2 lässt sich damit beispielsweise von zehn auf die minimale Anzahl von vier Relationen vereinfachen (siehe Abbildung 4-5). Generell erhält man für rechtwinklige Gebäudestrukturen mit n Flächen einen vollständigen Relationsgraph mit $\binom{n}{2} = \frac{n(n-1)}{2}$ Relationen, der auf das Minimalgerüst mit $n-1$ Relationen reduziert werden kann.

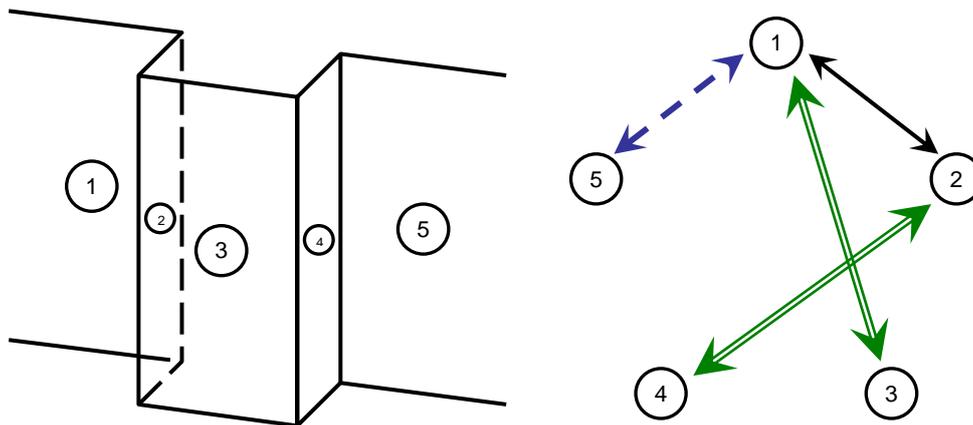


Abbildung 4-5: Fassadenausschnitt mit minimalen Gerüst des zugehörigen Relationsgraphen.

4.3 Formvereinfachung

Wie schon in Kapitel 3 diskutiert, dürfen bei der Generalisierung nicht einfach Eckpunkte oder Kanten aus dem Modell entfernt werden. Auch wenn der dadurch eingeführte geometrische Fehler klein ist, so können die geometrischen Relationen verletzt werden. Im Prinzip wäre es ausreichend die Gültigkeit der geometrischen Relationen nach jeder Vereinfachungsoperation zu prüfen. Sind nicht mehr alle Relationen nach der Operation erfüllt, so ist diese ungültig und darf nicht durchgeführt bzw. muss wieder rückgängig gemacht werden. Mit dieser Einschränkung wären die aus der Flächensimplifizierung bekannten Operatoren auch für die Generalisierung von 3D-Gebäudemodellen anwendbar. Das Problem ist jedoch, dass beispielsweise ein Edge Collapse Operator nicht mächtig genug ist um unter diesen Bedingungen ein Gebäudemodell zu vereinfachen. Abbildung 4-6 zeigt den Fassadenausschnitt mit vier möglichen Vereinfachungen. Einzeln ausgeführt sind diese Operationen nicht gültig, da sie sowohl die Planarität der Fläche nicht einhält als auch die Parallelitätsrelation zwischen der vorderen Fläche und den beiden hinteren Flächen verletzen. Wenn man diese vier Operationen jedoch zu einer neuen unteilbaren Operation verknüpft, so entsteht eine gültige Generalisierungsoperation.

Das Beispiel führt zu der Erkenntnis, dass man bekannte Simplifizierungsoperatoren durchaus für die Generalisierung von 3D-Gebäudemodellen nutzen kann, solange man mehrere Operationen geeignet zu einer komplexen Operation kombiniert. Die Schwierigkeit ist nun eine solche gültige Kombination zu finden.

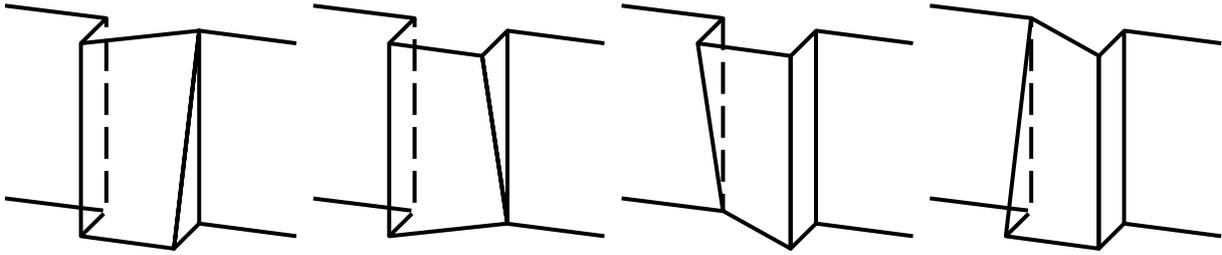


Abbildung 4-6: Die Ausbuchtung im Fassadenausschnitt kann nicht mit einer einzigen Edge Collapse Operation gültig vereinfacht werden.

4.3.1 Erkennung und Entfernung von Merkmalen

Ähnlich dem Ansatz von Staufenbiel [1973] wird bei der Erkennung und Entfernung von Merkmalen die geometrische Situation zunächst untersucht. Die Gebäudegeometrie wird dann mit Hilfe eines speziell auf die erkannte Merkmalsklasse abgestimmten Operators vereinfacht. Häufige Merkmale von 3D-Gebäudemodellen sind beispielsweise: Versatz, Ein- und Ausbuchtung, Ecke, Kerbe, Spitze etc. An Hand der Primitive, über die sich ein Merkmal definiert, kann man Merkmalsklassen unterscheiden. Der Versatz, die Ein- und Ausbuchtung und die Ecke sind Flächenmerkmale, während die Kerbe ein Kanten- und die Spitze ein Knotenmerkmal sind (siehe Abbildung 4-7).

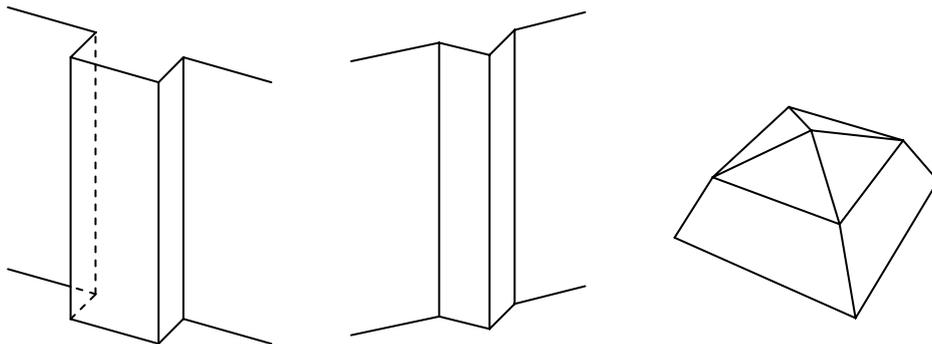


Abbildung 4-7 Beispiele für Flächen-, Kanten- und Knotenmerkmale.

Für die Erkennung von Merkmalen untersucht der Algorithmus nacheinander alle Flächen, Kanten und Knoten und testet, ob das Primitiv ein Merkmal definiert. Dafür muss pro Merkmalsklasse ein expliziter Regelsatz aufgestellt werden. Eine Fläche wird beispielsweise mit Hilfe der folgenden zwei Regeln sehr zuverlässig als Ausbuchtung erkannt:

1. Der Winkel zwischen dem Normalenvektor der Vorderseite und den Normalenvektoren der Seitenflächen muss sich innerhalb eines Toleranzbereichs um die 90° sein. Dieser Toleranzbereich sollte nicht zu groß gewählt werden. Ansonsten werden Ecken und Flächen, die Teil eines gekrümmten Elements sind, fälschlicherweise als Ausbuchtung erkannt.
2. Alle Kanten, die über einen Knoten an die Vorderfläche der Ausbuchtung grenzen aber nicht auch Kante dieser Fläche sind, müssen vollständig hinter der Vorderseite liegen.

Ein erkanntes Merkmal wird sogleich mit einem Wert versehen, der die Signifikanz des Merkmals für das Gebäudemodell quantitativ beschreibt. Für eine Ausbuchtung ist dies die minimale Länge aus der Menge der Kanten, welche die obige Regel 2 erfüllen. In jedem Vereinfachungsschritt wird nun das Merkmal mit der geringsten Signifikanz mittels eines merkmalsklassenspezifischen Operators aus dem Gebäudemodell entfernt.

Die Entfernung einer Ausbuchtung erfolgt durch eine Kombination zweier Operatoren und zwar des Edge-Collapse Operators und der Kantenverkürzung. Der Edge-Collapse Operator wurde schon in Kapitel 3 vorgestellt. Er entfernt eine Kante, indem er die beiden Endpunkte zu einem einzigen Punkt zusammenzieht.

Dabei verändern sich auch die an die Kante grenzenden Flächen. Die Kantenverkürzung bewahrt dagegen die Topologie des Modells. Ein Eckpunkt wird entlang der Kante in Richtung des anderen Punktes bewegt, wodurch sich ausschließlich die Kante verkürzt. Bei beiden Operatoren muss für konkave Polygone beachtet werden, dass die Kanten des Ergebnispolygons sich kreuzen können. Eine solche Operation ist nicht gültig und muss verhindert werden.

Eine Ausbuchtung wird aus dem Gebäudemodell entfernt, indem die Kanten, welche Regel 2 der Erkennung von Merkmalen erfüllen, wie folgt behandelt werden:

- Kanten, die gleichlang wie die kürzeste Kante sind, werden mittels Edge-Collapse Operator entfernt. Der Knoten, der an die Vorderfläche der Ausbuchtung grenzt, wird dabei in den Knoten, der an die hinteren Flächen grenzt, verschoben.
- Alle anderen Kanten werden um die Länge der kürzesten Kante in dieselbe Richtung verkürzt.

Abbildung 4-8 zeigt, wie die beiden Operatoren eine Ausbuchtung mit Versatz schrittweise entfernen. Der Verkürzungsoperator ist hier notwendig, da nicht alle an die Ausbuchtung grenzenden Flächen in derselben Ebene liegen. Ohne die Verkürzung würde die Planarität der Vorderseite nicht erfüllt.

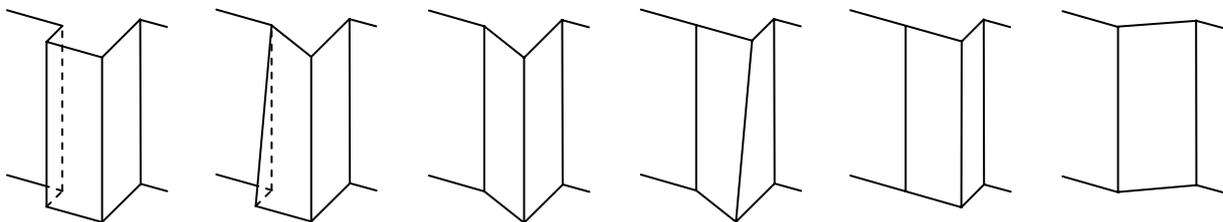


Abbildung 4-8: Das Entfernen von beliebigen Ausbuchtungen gelingt nur mittels der Kombination aus Edge-Collapse- und Verkürzungsoperationen. Ohne die Verkürzung von Kanten würden unerwünschte Schrägflächen entstehen (letztes Bild).

Wird ein Merkmal aus dem Gebäudemodell entfernt, so muss anschließend der Relationsgraph angepasst werden. Wie schon weiter oben erwähnt, werden die inzidierenden Relationskanten gelöscht, sobald eine Fläche bzw. ein Knoten aus dem Graphen entfernt wurde. Des Weiteren wird überprüft, ob Parallelitätsrelationen in Koplanaritätsrelationen umgewandelt werden können. Abschließend werden die Relationen des Relationsgraphen überprüft. Nur, wenn alle Relationen noch gültig sind, ist die Entfernung des Merkmals eine gültige Operation und darf übernommen werden.

Bei der Anwendung des Edge-Collapse Operators muss beachtet werden, dass dieser eine lokale Invertierung des Dreiecks- bzw. Polygonnetzes verursachen kann. Dabei ändert sich die Orientierung der Flächenprimitive. Um dieses Umschlagen der Geometrie zu verhindern, wird getestet, ob die Operation die Orientierung der angrenzenden Flächenprimitive ändert oder anderweitig ungültig macht. Jede solche Operation wird verworfen. Die Vermeidung von Invertierungen durch eine explizite Kontrolle der Orientierung ist weit verbreitet und ist in [Garland und Heckbert 1997], [Hoppe 1996] und [Ronfard und Rossignac 1996] für die Behandlung von Dreiecksnetzen sowie in [Stadt und Gross 1998], [Trotts et al. 1999] und [Cignoni et al. 2000] für Tetraedervolumen dokumentiert.

4.3.2 Innere Punkte

Die Knotenpunkte des originalen Gebäudemodells werden bei der Geometrievereinfachung nicht einfach verändert oder entfernt. Vielmehr werden die ursprünglichen Positionen mit den Knotenpunkten oder Flächen als so genannte innere Punkte gespeichert, die bei der Vereinfachung erhalten bleiben. Für das Beispiel in Abbildung 4-8, der Entfernung einer Ausbuchtung, ergeben sich folgende Konsequenzen:

- Die durch Edge Collapse Operationen gelöschte Knotenpositionen werden mit der Frontfläche der Ausbuchtung gespeichert. Werden zwei oder mehr koplanare Flächen vereinigt, so vereinigen sich auch die Mengen der inneren Punkte.

- Die ursprünglichen Positionen der durch Kantenverkürzung veränderten Eckpunkte werden in den entsprechenden Knoten gespeichert.
- Innere Punkte werden nicht ersetzt oder verändert, auch wenn die assoziierte Fläche oder der Eckpunkt sich durch mehrfache Vereinfachungsoperationen weiter ändern. Dadurch wird gewährleistet, dass ein innerer Punkt immer die Ausgangsposition eines Eckpunkts speichert.

Dadurch bleiben die Punktinformationen im Modell erhalten und es wird zusätzlich eine Zuordnung dieser Punkte zu den Flächen bzw. Knoten des generalisierten Gebäudemodells durchgeführt. Die inneren Punkte werden, wie man im nächsten Abschnitt sehen wird, in der Formeinpassung mittels Ausgleichung nach kleinsten Quadraten berücksichtigt.

4.4 Formeinpassung

Durch die geometrische Vereinfachung entfernt sich das Gebäudemodell von den originalen Punkten des Eingabemodells. Werden beispielsweise nur Ausbuchtungen entfernt, so ist das Ergebnismodell anschließend kleiner als das Ausgangsmodell (siehe Abbildung 4-9). Um dem entgegenzuwirken, findet eine Formeinpassung an die originalen Eingabepunkte mittels einer Ausgleichung nach kleinsten Quadraten statt.

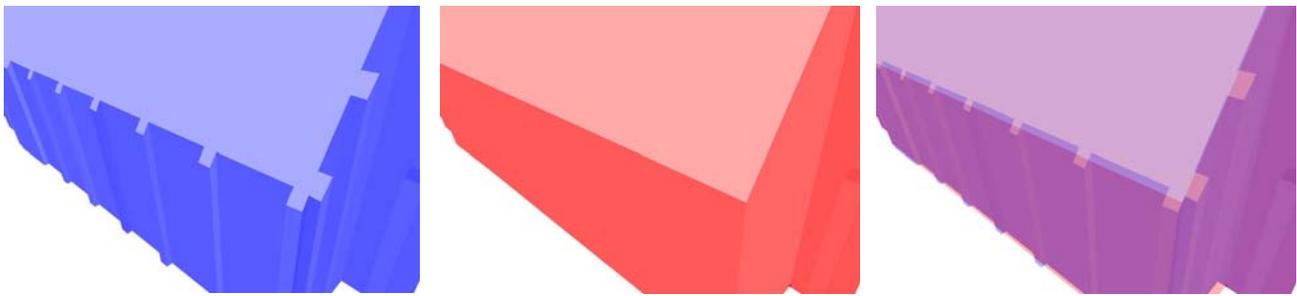


Abbildung 4-9: Gebäudemodell mit Ausbuchtungen (links), generalisiertes Modell nach Formeinpassung (Mitte) und Überlagerung beider Modelle (rechts).

Die Parameterschätzung der Ausgleichung erfolgt im gemischten Modell nach Gauß-Helmert oder auch „Bedingte Beobachtungen mit Unbekannten“ [Wolf 1975] genannt. Die Modelldefinition lautet:

$$Av + Bx + w = 0$$

$$D(l) = \sigma^2 P^{-1} := D(v)$$

Ziel ist es, die quadratische Funktion $\|v\|_p^2 = v'Pv = \min$ unter $Av + Bx + w = 0$ zu minimieren. Das bedeutet, dass die minimale Quadratsumme der Verbesserungen der linearisierten Beobachtungen B unten den Bedingungen B minimiert wird. Deren Normalgleichungen lauten:

$$AP^{-1}A'\hat{\lambda} + B\hat{x} + w = 0$$

$$B'\hat{\lambda} = 0$$

Durch Auflösung der ersten Gleichung nach $\hat{\lambda}$ und Einsetzen in die zweite Gleichung folgt:

$$\hat{x} = -(B'\tilde{P}^{-1}B)^{-1}B'\tilde{P}^{-1}w \text{ mit } \tilde{P} = AP^{-1}A'$$

Das vereinfachte Gebäude wird im funktionalen Modell als eine Menge von Ebenen und deren Schnittpunkte beschrieben. Ein Würfel würde beispielsweise durch sechs Ebenen und acht Schnittpunkte definiert. Zusätzlich gehen in die Ausgleichung die geometrischen Relationen als Bedingungen mit ein. Durch die Ausgleichung werden die Positionen der Schnittpunkte unter Berücksichtigung dieser Zwänge optimiert. Die Verwendung von Ebenen garantieren letztendlich, dass die Schnittpunkte einer Fläche planar im Raum angeordnet werden.

Eine Ebene wird durch die folgende Gleichung beschrieben:

$$E_i: A_i x + B_i y + C_i z + D_i = 0$$

Dabei ist der Vektor (A_i, B_i, C_i) der Normalenvektor der Ebene und D_i ist der kleinste Abstand zum Ursprung des Koordinatensystems. Die Ebenenparameter werden durch die Positionen der nicht veränderten Eckpunkte und die inneren Punkte aller auf der Ebene liegenden Flächen bestimmt. Die zu einer Ebene gehörigen Flächen können sehr einfach im Relationsgraph identifiziert werden, da diese durch Kante für die **Koplanarität** miteinander verbunden sind. Es wird auch hier nicht unterschieden, ob Flächen entgegengesetzt orientiert sind. Stattdessen werden die Ebenen anfänglich so normiert, dass der vom Betrag höchste Parameter einen positiven Wert erhält. Dadurch wird gewährleistet, dass parallele Ebenen dieselben Parameter A, B und C haben. Aus k Punkten ergeben sich für die Ebene E_i die folgenden Beobachtungen:

$$E_i: A_i x_k + B_i y_k + C_i z_k + D_i = 0$$

Da die Definition der Ebenenparameter überparametrisiert ist, wird die Normierung des Vektors A_i, B_i, C_i als zusätzliche Bedingungen in die Ausgleichung integriert. Somit wird die Triviale Lösung vermieden, bei der alle Parameter den Wert Null annehmen:

$$A_i^2 + B_i^2 + C_i^2 = 1$$

Die geometrischen Relationen Parallelität und Rechtwinkligkeit von Flächen bzw. Ebenen aus dem Relationsgraph werden ebenfalls mit Bedingungen in den Ausgleichungsprozess integriert. Das minimale Gerüst des Graphen stellt sicher, dass die Flächen einer Ebene nur durch genau eine Parallelitäts- oder Rechtwinkligkeitskante mit Flächen einer anderen Ebene verbunden sind.

Parallelität: Für n parallele Ebenen E_1 bis E_n werden dieselben Parameter A, B und C in den Beobachtungsgleichungen verwendet. Das bedeutet, dass die Parameter A, B und C über alle parallelen Flächen geschätzt werden, während sich nur der Parameter D für jede einzelne Ebene unterscheidet.

$$E_1: Ax + By + Cz + D_1 = 0$$

$$E_n: Ax + By + Cz + D_n = 0$$

Rechtwinkligkeit: Die Zwangsbedingung für jeweils zwei rechtwinklige Ebenen E_1 und E_2 wird aus der Gleichung zur Berechnung deren Schnittwinkel wie folgt abgeleitet:

$$\cos \phi = \frac{A_1 A_2 + B_1 B_2 + C_1 C_2}{\sqrt{A_1^2 + B_1^2 + C_1^2} \sqrt{A_2^2 + B_2^2 + C_2^2}}$$

Da der Kosinus bei 90° den Wert Null hat, lässt sich die Gleichung auf den Zähler reduzieren und ergibt die folgende Bedingungsgleichung:

$$A_1A_2 + B_1B_2 + C_1C_2 = 0$$

Die Ausgleichung hat zum Zweck, die Knotenpunkte des ausgeglichenen Modells auf die Schnittpositionen der Ebenen zu ziehen und dabei die geometrischen Relationen zu berücksichtigen. Die Relationen sind über die oben definierten Zwangsbedingungen integriert. Jetzt müssen noch die Schnittpunkte als Beobachtungen mit aufgenommen werden, da es ansonsten vorkommen kann, dass sich bei vier oder mehr Ebenen kein eindeutiger Schnittpunkt ergibt. Die Vorgehensweise wird zusätzlich motiviert, dass bisher die topologischen Informationen des Gebäudemodells noch nicht berücksichtigt werden.

Die Koordinaten X_i , Y_i und Z_i des Schnittpunkts P_i werden bestimmt, indem für jede Ebene E_k je eine weitere Beobachtungsgleichung in die Ausgleichung mit aufgenommen wird:

$$P_i: A_kX_i + B_kY_i + C_kZ_i + D_k = 0$$

Eine unterschiedliche Gewichtung der Bedingungstypen erlaubt eine Einflussnahme auf das Ergebnis um beispielsweise eindeutige Schnittpunkte über strikt parallele Gebäudeflächen zu favorisieren.

Der Ausgleichungsprozess passt das generalisierte Gebäudemodell unter Beachtung der Bedingungsgleichungen möglichst gut in die originalen Punktpositionen ein. Aufgrund der dabei durchgeführten Minimierung der Quadratsumme werden kleine Fehler in Kauf genommen um dadurch große Fehler zu verhindern. Das vereinfachte Beispielgebäude aus Abbildung 4-10 wurde nur auf einer der vier Fassadenseiten verändert. Die rechtwinkligen Fassadenseiten sowie die Grundseite sind mittels Bedingungen für Parallelität und Rechtwinkligkeit im Relationsgraph miteinander verknüpft. Man würde erwarten, dass die vereinfachte Fassadenseite durch die Formeinpassung parallel nach außen verschoben wird und zwischen der ursprünglichen und jetzigen Position zum liegen kommt. Durch die Ausgleichung drehen sich jedoch diese vier Seiten. Dies geschieht um die vereinfachte Fassadenseite besser an die sechs Punkte der originalen drei Polygonflächen anzupassen. Das hat zur Folge, dass sich die Lage der drei unveränderten Seiten verschlechtert. Da zwischen den beiden Dachebenen und den Fassadenebenen keine Relationen bestehen, bleiben die beiden Firstpunkte an ihrer ursprünglichen Position. Dies bewirkt eine asymmetrische und schiefe Dachform.

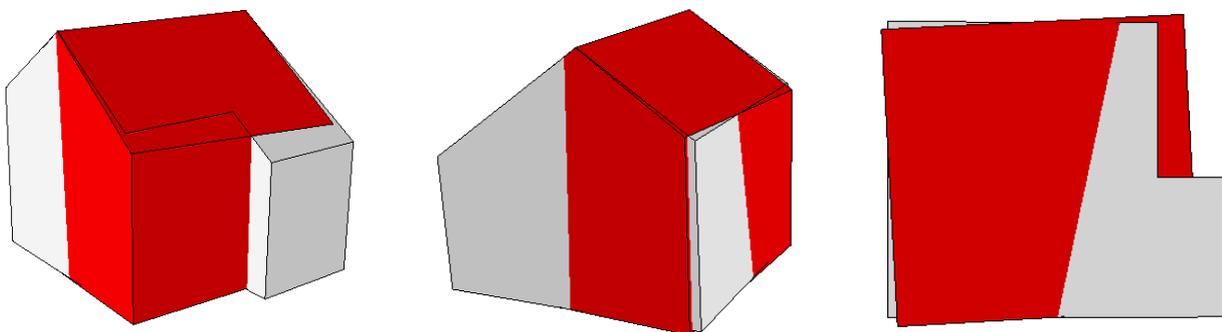


Abbildung 4-10: Originalmodell (grau) und durch Formeinpassung verdrehtes, generalisiertes Modell (rot).

Um die Drehung zu verhindern, kann man zusätzliche, feststehende Ebenen einfügen. An diesen können die freibeweglichen Ebenen mittels Bedingungen ausgerichtet werden. Das Ergebnis in Abbildung 4-11 wurde beispielsweise über zwei solche Ebenen realisiert, welche entlang der beiden Hauptachsen durch den Ursprung des Gebäudes verlaufen. Dadurch sind die vier Fassadenseiten des Gebäudemodells nur noch parallel

verschiebbar. Durch die Formeinpassung hat sich nur die vereinfachte Seite parallel nach außen verschoben. Alle anderen Flächen blieben unverändert, so dass auch die regelmäßige Dachstruktur erhalten blieb. Weil sich die eine Dachfläche verlängert hat, besitzt das Satteldach nun jedoch zwei unterschiedliche Traufhöhen.

Da die Generalisierung von 3D-Gebäudemodellen ein noch sehr junges Forschungsgebiet ist, gibt es noch keine Erkenntnis darüber wie das Ergebnis auszusehen hat. Das Problem ist, dass keine Abbildungsvorschriften existieren, wie dies bei der 2D-Generalisierung im Laufe der Jahre geschehen konnte. Es stellt sich die Frage, wie eine optimale Generalisierung eines 3D-Gebäudemodells aussehen soll. Man würde vielleicht erwarten, dass der Dachfirst parallel um die halbe Fassadenseite mitwandert, um ein symmetrisches Satteldach zu erzeugen. Dies ist jedoch nur mit weiteren Bedingungen in der Formeinpassung zu realisieren. Des Weiteren müsste schon im Vorfeld eine Interpretation der gesamten Gebäudestruktur existieren. Dies kann man für so einfache Dachformen durch eine der Formvereinfachung vorgeschaltete Analyse bewerkstelligen. Für allgemeine Formen ist diese jedoch nicht ohne weiteres möglich.

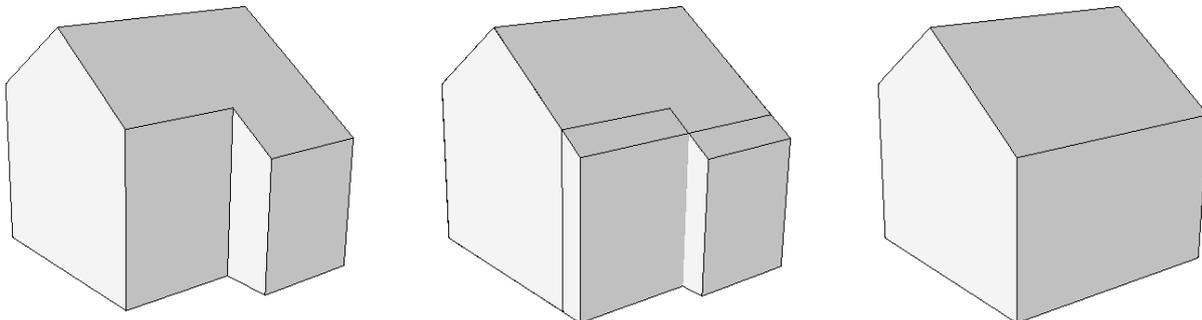


Abbildung 4-11: Original 3D-Gebäudemodell mit einfachem Schleppdach (links), generalisiertes Ergebnismodell mit Satteldach (rechts) und die Überlagerung beider Modelle (Mitte).

4.5 Texturen

Für texturierte 3D-Gebäudemodelle stellt sich die Frage, wie während des Generalisierungsprozesses mit den Texturinformationen umgegangen werden soll. Geht man davon aus, dass die Texturbilder nicht verändert werden sollen, so bleibt nur die Modifikation der Texturkoordinaten. Für die Entfernung von Ausbuchtungen ergibt sich folgende Situation: Die Texturinformation der Vorderseite kann unverändert übernommen werden, da sich keine Änderungen am Polygon ergeben. Werden die Seitenflächen durch die Operation vollständig aus dem Modell entfernt, so wird die Vorderfläche parallel in die Ebene der hinteren Polygonflächen geschoben. Die Texturbilder der Seitenflächen können einfach mit der Geometrie aus dem Modell entfernt werden ohne das Erscheinungsbild stark zu stören. Die fehlenden Seitenbilder kleiner Vorsprünge fallen zumeist gar nicht auf (siehe Abbildung 4-14). Da sich Geometrien nur bedingt bildbasiert simulieren lassen, erscheinen große Vorsprünge wie beispielsweise Balkone auf der nun flachen Fassade zwar bei naher Betrachtung ungewohnt, aber immer noch akzeptabel. Die Bildsynthese der Seiten- und Vorderflächen zu einer neuen statischen Textur ist maximal für eine Schrägansicht möglich und wirkt für alle anderen Ansichten schlichtweg falsch. Werden die Seitenflächen nicht entfernt sondern nur verkürzt, so sind die neuen Texturkoordinaten der sich ändernden Eckpunkte entlang der Kontraktionskanten durch lineare Interpolation neu zu berechnen. Dadurch werden Teile der Textur abgeschnitten, was zur Folge haben kann, dass Wände inmitten eines Fensters oder ähnlichem abschließen. Ohne eine Bildanalyse des Texturinhalts kann dies jedoch nicht ohne weiteres verhindert werden. Alternativ könnte man die Texturkoordinaten auf ihren ursprünglichen Wert belassen um den gesamten Bildinhalt darzustellen, was jedoch eine ungleichmäßige Stauchung innerhalb einer Fassadenseite zur Folge hätte.

Ein weiteres Problem ist die Formeinpassung an die originalen Eingabepunkte. Hierbei wird das vereinfachte Gebäudemodell in unterschiedliche Richtungen gestaucht und gedehnt. Da sich die geometrischen Veränderungen aber regelmäßig auf das gesamte Gebäude auswirken und im Vergleich zur Größe des Modells auch nur gering ausfallen, kann man diese Auswirkungen normalerweise vernachlässigen. Sollen jedoch die absoluten Dimensionen der Fassadenobjekte (Türen, Fenster, Dachrinne usw.) beibehalten werden, so ist eine Synthese der Bildinhalte der vorhandenen Texturbilder notwendig. Hierzu müssen die Fassadenobjekte von

den Wandbereichen getrennt und auf einer synthetisierten Wandtextur mit den entsprechenden Dimensionen neu angeordnet werden. Dies ist für beliebige Fassadenstrukturen jedoch automatisch zurzeit nicht möglich.

4.6 Ergebnisse

Der Generalisierungsalgorithmus wurde implementiert und für komplexe Gebäudemodelle ausgeführt. Die Form der Gebäude wurde mit der beschriebenen Methode zur Erkennung und Entfernung von Ausbuchtungen vereinfacht. Die anschließende Formeinpassung konnte jedoch nur für Teile des Gebäudes durchgeführt werden, da für die zur Verfügung stehenden Datensätze erst eine manuelle Nachbearbeitung schräger Flächen notwendig war. Diese Ungenauigkeiten folgen aus der semi-automatischen Gebäuderekonstruktion, bei der offensichtlich die geometrischen Relationen bei der Erfassung nicht explizit berücksichtigt wurden.

Die Ergebnisse sollen exemplarisch an den drei Gebäudemodellen aus Abbildung 4-12 aufgezeigt werden. Die Modelle wurden ausgewählt, da sie zu den komplexeren des gesamten 3D-Stadtmodells von Stuttgart gehören und auch über eine größere Anzahl von Vorsprüngen verfügen. Ein weiteres Auswahlkriterium war die Verfügbarkeit von realen Gebäudetexturen. Der visuelle Eindruck kann daher nicht nur aufgrund der geometrischen Vereinfachung beurteilt werden, sondern auch als Ergebnis der fehlenden Bildinformationen.



Abbildung 4-12: Gebäudemodelle vom Neuen Schloss von Stuttgart (links), der Stiftskirche (Mitte) und der Markthalle (rechts).

Für das Modell des Neuen Schlosses von Stuttgart (siehe Abbildung 4-13) konnten in drei Iterationen insgesamt 110 Ausbuchtungen gefunden und entfernt werden. Dies hat eine Reduktion der Komplexität von 2730 auf 1837 Dreiecke zum Ergebnis.



Abbildung 4-13: Hintere Fassadenseite des Neuen Schloss vor (links) und nach (rechts) der Generalisierung.

Abbildung 4-14 zeigt einen Eckbereich des Neuen Schlosses von Stuttgart. Obwohl die Flächen des Ausgangsmodells nicht strikt parallel und rechtwinklig sind, wurden geometrische Relationen definiert, die im generalisierten Modell durch die Formeinpassung erzwungen wurden.



Abbildung 4-14: Eckbereich des Neuen Schlosses von Stuttgart vor (links) und nach (rechts) der Generalisierung.

Die Stiftskirche von Stuttgart besaß vor der Generalisierung 594 Dreiecke, von denen nach der Entfernung von insgesamt 24 Vorsprüngen noch 388 Dreiecke verblieben (siehe Abbildung 4-15). Im Gebäudemodell der Markthalle konnten sieben Vorsprünge identifiziert werden, durch deren Entfernung sich die Komplexität des Gebäudemodells von 848 Dreiecken auf 798 Dreiecke reduzierte (siehe Abbildung 4-16). Das Modell besitzt noch weitere 14 Vorsprünge, die jedoch aufgrund ihrer stark abgerundeten Form nicht automatisch als Vorsprünge identifiziert werden konnten.

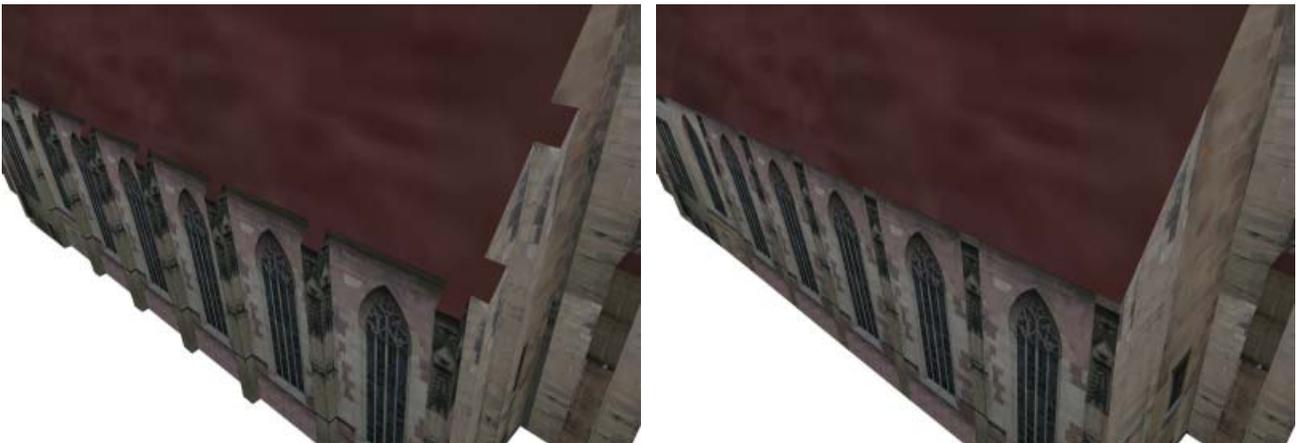


Abbildung 4-15: Gebäudemodell der Stiftskirche vor (links) und nach (rechts) der Generalisierung.



Abbildung 4-16: Markthalle vor (links) und nach (rechts) der Generalisierung.

4.7 Diskussion

Durch die Vereinfachung der Ausbuchtungen wurden die Seitenflächen komplett eliminiert, wodurch sich die Vorderflächen geometrisch gut in die Fassaden einfügen konnten. Lediglich deren Texturen zeigen die ursprüngliche Fassadenstruktur an. Da jedoch nur die Texturen übrig bleiben, die auch frontal sichtbar sind, wird das Erscheinungsbild soweit nicht gestört. Bei Darstellungen ohne Texturen sind keine Artefakte zu erkennen. Insbesondere bei der Vereinfachung des Neuen Schlosses kam gezielt der Kantenverkürzungsoperator zum Einsatz, da auf einer Seite der Gebäudeecke sehr lange Kanten vorkommen. Ohne die Verkürzung wäre es nicht möglich, die Ausbuchtung an der Ecke ohne Verletzung geometrischer Relationen zu entfernen. Allgemein bedeutet dies, dass der vorgestellte Operator einen Versatz, der sich nahe einer Gebäudeecke befindet, oder Z-Strukturen, die sich über die gesamte Fassadenseite erstrecken, korrekt als eine spezielle Form der Ausbuchtung klassifizieren und entfernen kann. Auch Fassadenstrukturen mit gestaffelten Ausbuchtungen werden durch die iterative Vorgehensweise schrittweise vereinfacht. Allein durch das Entfernen von Ausbuchtungen konnte die Detaillierung der Modelle schon deutlich reduziert werden. Problematisch haben sich jedoch Ausbuchtungen herausgestellt, die sehr stark abgerundet sind. Hier gilt es noch weitere Merkmale zu definieren und für diese geeignete Operatoren zu entwickeln. Es fehlen auch Operatoren zur Erzeugung von Blockgebäuden, wie sie beispielsweise von CityGML benötigt werden. Hierzu wäre es denkbar die Orthogonalisierung von Dachflächen zu nutzen, wie sie von Forberg und Mayer [2003] vorgestellt wurden (siehe Abschnitt 3.2).

Die geometrischen Relationen von 3D-Gebäudemodellen werden eingehalten oder konnten, wie im Falle des Neuen Schlosses, erzwungen werden. Ein Problem stellen jedoch Kanten dar, die nicht mittels Bedingungen „zusammengehalten“ werden. Durch die Ausgleichung nach kleinsten Quadraten kann es passieren, dass bei besonders stumpfen oder spitzen Winkeln die Nachbarflächen ausreißen und sich erst weit entfernt oder im schlimmsten Fall gar nicht schneiden. Solche geometrischen Situationen kommen besonders bei der Approximation runder Türme mittels planarer Flächen vor. Dafür wäre es vielleicht sinnvoll für die Formeinpassung weitere Zwangsbedingungen einzuführen, die dem entgegenwirken.

Die Anzahl unterstützter Merkmale und geometrischer Relationen haben einen deutlichen Einfluss auf die Verarbeitungszeit der Generalisierung. Das Verfahren ist daher als Stapelverarbeitung gedacht, die vor der eigentlichen Anwendung abläuft und die generierten Modelle in einer Datenbank abspeichert. Die verwendeten Simplifizierungsoperatoren erlauben zusätzlich die Speicherung einer Generalisierungshistorie wie beispielsweise bei den Progressiven Netzen. Es ist daher möglich Modelle in verschiedenen Generalisierungsstufen bedarfsgerecht und zeitnah aus der Historie abzuleiten, so dass diese auch für kartenverwandte Echtzeitdarstellungen genutzt werden können.

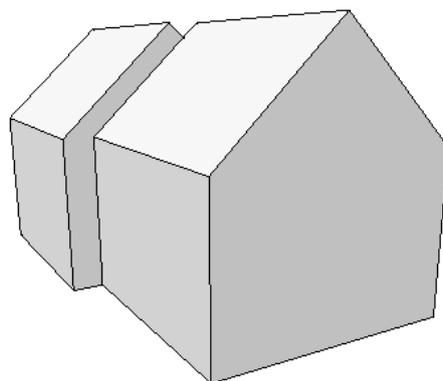


Abbildung 4-17: Zwei Doppelhaushälften mit Satteldach

Zusammenfassend kann man sagen, dass es mit dem beschriebenen Generalisierungsverfahren möglich ist die Detaillierung von 3D-Gebäudemodellen zu reduzieren und dabei die geometrischen Relationen zu bewahren. Durch die Kombination mehrerer Edge-Collapse- und Kantenverkürzungsoperationen zu neuen Operatoren werden Merkmale konsistent aus dem Modell entfernt. Es handelt sich daher um ein lokales Verfahren, welches die globale Struktur des Gebäudes nicht in die Verarbeitung mit einbezieht. Dadurch ist aber

auch die Limitierung des Verfahrens gegeben. Beispielsweise werden sehr komplexer Regeln notwendig sein, die Struktur zweier angrenzender Doppelhaushälften als Merkmal zu identifizieren und konsistent zu entfernen, so dass anschließend ein langes Gebäude mit einem einheitlichen Satteldach entsteht (siehe Abbildung 4-17). Daher wird im nächsten Kapitel ein globales Generalisierungsverfahren vorgestellt, welches unter Berücksichtigung der Gesamtstruktur auf Basis des Originalmodells ein vereinfachtes Modell nachbaut.

5 Generalisierung zur globalen Formvereinfachung

Im vorhergehenden Kapitel wurde ein Generalisierungsverfahren vorgestellt, welches 3D-Gebäudemodelle durch sukzessives Entfernen kleiner Strukturelemente geometrisch vereinfacht. Nun soll ein komplementärer Ansatz beschrieben werden, welcher das Modell nicht mehr lokal betrachtet sondern bei der vereinfachten Gebäudeform auf globaler Ebene herausgearbeitet wird. Ausgehend von einem 3D-Gebäudemodell, welches als Polyeder in Randbeschreibung repräsentiert ist, erfolgt eine Nachmodellierung des Objekts mittels Zellenzerlegung. Die Vereinfachung ergibt sich durch die Dekomposition aus Ebenen, welche die Ursprungsgeometrie nicht exakt modelliert sondern nur mehr approximiert. Diese Vorgehensweise begründet sich darauf, dass Gebäudefassaden, auch wenn sie sehr detailliert ausgearbeitet sind, bei der Generalisierung häufig durch ebene Flächen ersetzt werden können, ohne die charakteristische Gebäudeform zu zerstören.

5.1 Generalisierung mittels Zellenzerlegung

Bei der allgemeinen Zellenzerlegung ist die Raumteilung mittels unterschiedlicher Teilungsgeometrien, wie beispielsweise auch gekrümmte Flächen oder Halbkugeln, möglich. Der vorgestellte Ansatz beschränkt sich jedoch ausschließlich auf die Verwendung von Ebenen. Diese müssen in einem ersten Schritt des Algorithmus zunächst aus den originalen Gebäudedaten abgeleitet werden. Könnte eine ausreichende Anzahl an Ebenen gefunden werden (es werden mindestens vier Ebenen für die Repräsentation eines Tetraeders benötigt), so wird anschließend der zunächst unendliche Raum entlang der Ebenen in konvexe Halbräume aufgeteilt. Das generalisierte Gebäudemodell ist nun die Kombination derjenigen Primitiven, welche als zum Gebäude zugehörig identifiziert werden. Das generalisierte Gebäude liegt nun in einer Form vor, wie man es aus der Modellierungsform der Zellenzerlegung kennt. Durch die Boolesche Vereinigungsoperation können die Gebäudeprimitive schließlich zu einem Gesamtmodell zusammengefügt werden.

Zum besseren Verständnis soll die Verfahrensweise der Generalisierung an einem einfachen Beispiel im zweidimensionalen Raum demonstriert werden. Das Ziel ist die geometrische Vereinfachung des Grundrisspolygons aus Abbildung 5-1. Hierfür werden zunächst die Kanten des Polygons zu Geraden erweitert. Dabei bekommt jede Gerade eine Gewichtung, die sich aus der Länge der ursprünglichen Kante ergibt. Da kollineare Kanten dieselben Geraden erzeugen, werden diese zusammengefasst und deren Gewichtung aufsummiert. Aus der Menge der Geraden wird nun eine Zellenzerlegung des unendlichen Raums erzeugt. Da dies in der Praxis nur schwer realisierbar ist, reicht auch ein Minimalumschließendes Rechteck vom Grundrisspolygon mit der doppelten Größe. Anschließend werden die Zellen anhand ihrer Überdeckung mit dem Originalgrundriss in Gebäude- und Nichtgebäudezellen klassifiziert. Die Vereinigung der Gebäudezellen ergibt wieder das ursprüngliche Grundrisspolygon.

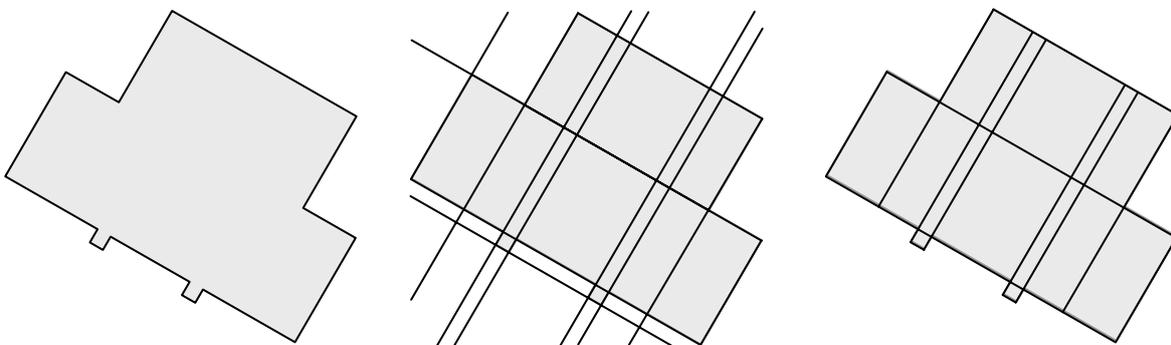


Abbildung 5-1: 2D-Gebäudegrundriss (links), Zerlegung des unendlichen Raums entlang der Polygonkanten (Mitte) und daraus resultierende Zellenzerlegung des Gebäudegrundrisses (rechts).

Bis zu diesem Punkt bleibt der Gebäudegrundriss jedoch noch unverändert. Anhand der resultierenden Zellenzerlegung des Grundrisses erkennt man, dass die Komplexität hauptsächlich durch kleinflächige Zellen entsteht. Um diese kleinflächigen Zellen möglichst gar nicht erst zu erzeugen, muss die Anzahl der Schnittgeraden verringert werden. Zu diesem Zweck werden die Geraden zu Puffern erweitert, was bedeutet, dass die Geraden sich parallel in beide Richtungen bis zu einer maximalen Breite ausdehnen können (siehe Abbildung 5-2). In einem iterativen Prozess werden nun die Geraden in absteigender Reihenfolge ihrer Gewichtung vergrößert. Dabei werden Kanten, die sich komplett innerhalb eines Pufferbereichs befinden, der entsprechenden Gerade zugeordnet. Der Puffer der zugeordneten Kante wird dabei gelöscht. Nach der Vergrößerung wird für die verbleibenden Geraden je eine neue Schnittgerade aus den sich im Pufferbereich befindlichen Kanten gemittelt.

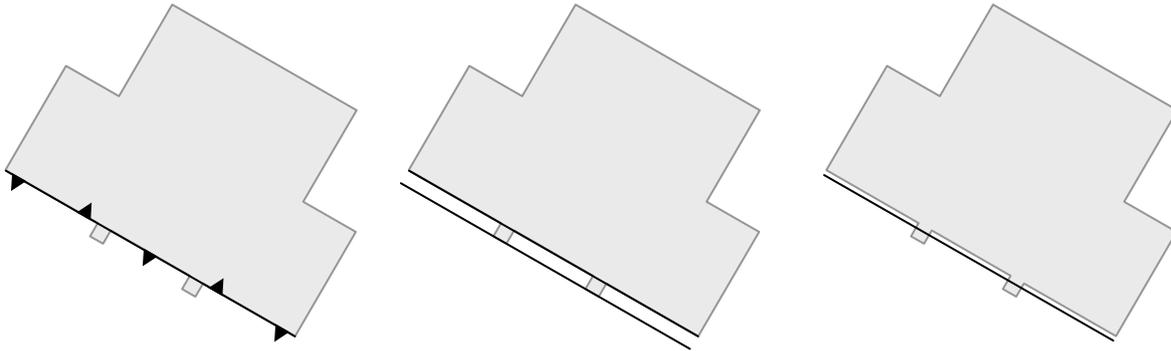


Abbildung 5-2: Puffer aus Grundrisslinien (links), maximierter Pufferbereich (Mitte) und aus den eingeschlossenen Liniensegmenten gemittelte Schnittgerade (rechts).

Die Verschneidung des unendlichen Raums bzw. des Minimalumschließenden Rechtecks doppelter Größe mit den gemittelten Geraden ergibt Zellen, die nicht mehr hundertprozentig deckungsgleich mit dem Grundriss sind. Daher wird für jede Zelle deren prozentuale Überdeckung mit dem Originalgrundriss berechnet. Vor allem für die Randzellen ergibt sich dabei nur ein sehr geringer Wert. Der vereinfachte Grundriss entsteht nun durch Zusammenfügen der Zellen mit sehr hoher Überdeckung (siehe Abbildung 5-3).

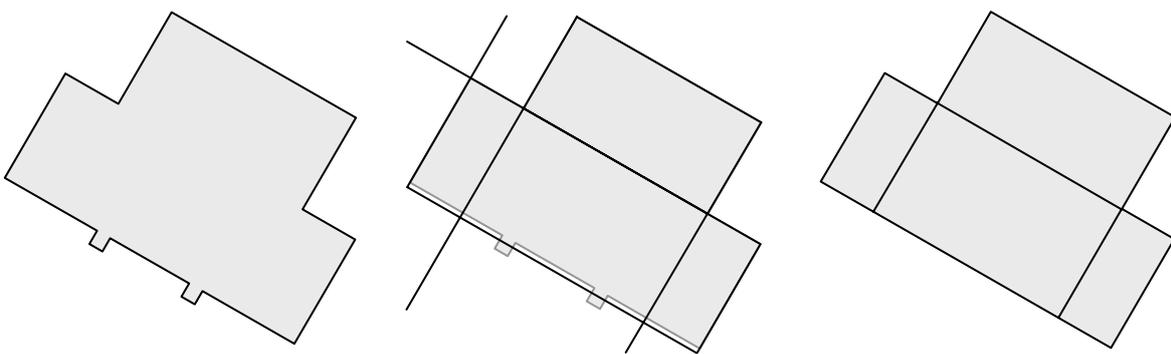


Abbildung 5-3: 2D-Gebäudegrundriss (links), Zerlegung des unendlichen Raums entlang der gemittelten Geraden (Mitte) und daraus resultierende vereinfachte Zellenzerlegung (rechts).

Die Generalisierung von 3D-Gebäudemodellen basiert auf einer leicht abgewandelten Form der beschriebenen 2D-Zellenzerlegung. Statt Schnittgeraden werden aus den Fassadenflächen Schnittebenen berechnet. Dabei werden auch Fassadenflächen innerhalb des Gebäudeumrings mit einbezogen, wodurch zusätzliche Schnitte an den Höhendiskontinuitäten des Dachs entstehen. Ohne diese Schnitte würden beispielsweise Türme im generalisierten Modell verschwinden. So kann für jede Zelle eine eigene Dachhöhe sowie Dachform berechnet werden. In der vorliegenden Arbeit werden zwei unterschiedliche Verfahren zur Erzeugung dieser Dachformen beschrieben. Das erste Verfahren generiert die Dachformen aus einer 3D-Zellenzerlegung und ähnelt von der Vorgehensweise der beschriebenen 2D-Grundrissgeneralisierung. Im zweiten Verfahren werden die Dachformen wie bei der Gebäuderekonstruktion mittels parametrisierter Stan-

Standardformen erzeugt. Nach der Konstruktion der Dachformen werden die 3D-Zellen zu einem Gesamtgebäudemodell zusammengeklebt. Zusammenfassend ergeben sich für die Generalisierung von 3D-Gebäudemodellen fünf Schritte (vgl. Abbildung 5-4), die in den nachfolgenden Abschnitten genauer erläutert werden.

Originalmodell

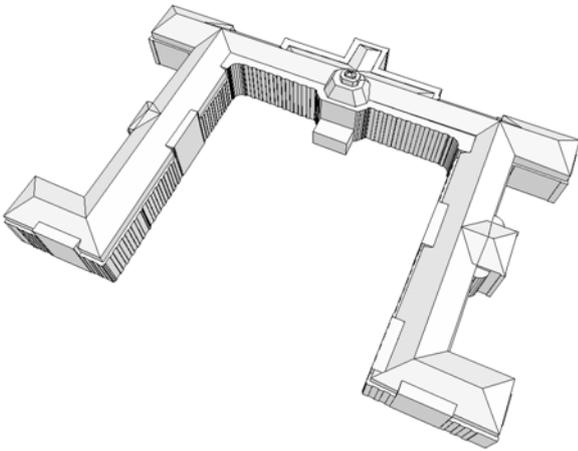
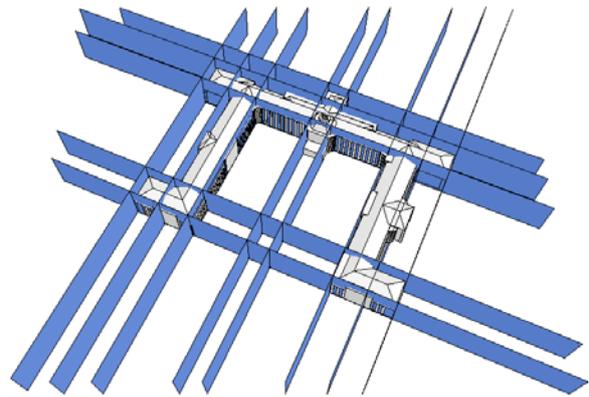
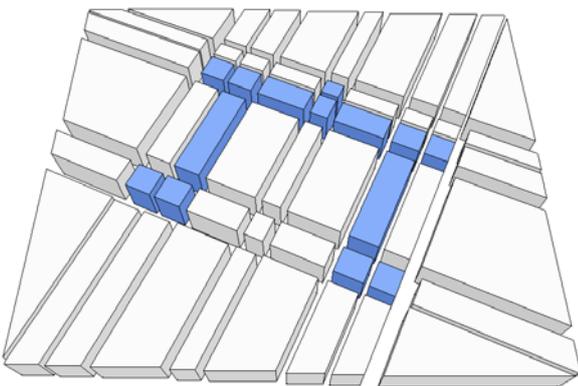
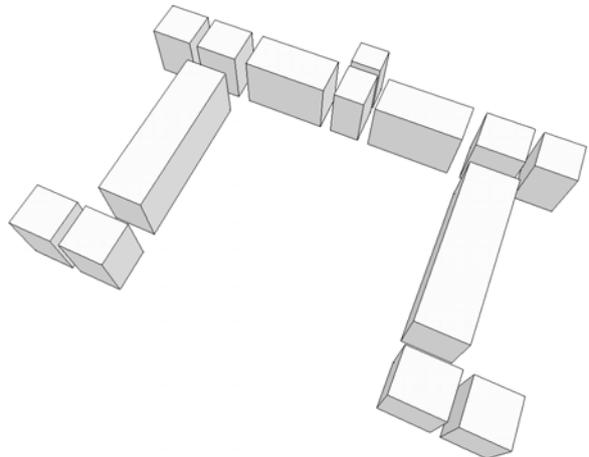
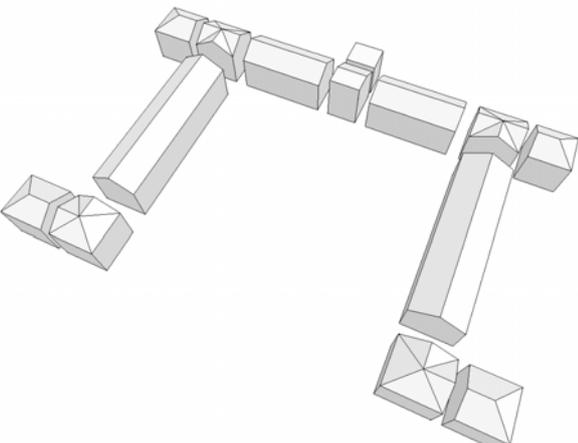
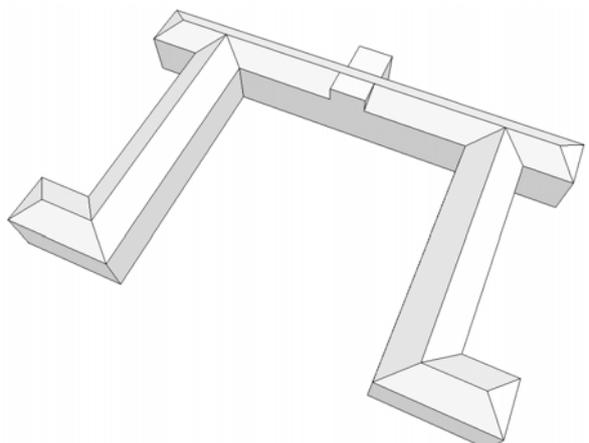
**Schritt 1: Ableitung von Schnittebenen**
(Abschnitt 5.2)**Schritt 2: Erzeugung der 2.5D-Zellerlegung** (Abschnitt 5.3)**Schritt 3: Identifikation der Gebäudezellen**
(Abschnitt 5.4)**Schritt 4: Generierung der 3D-Dachformen** (Abschnitt 5.5)**Schritt 5: Vereinigung der 3D-Gebäudezellen**

Abbildung 5-4: Generalisierungsschritte der globalen Formvereinfachung.

5.2 Ableitung von Teilungsebenen

Der Aufbau der Zellenzerlegung ergibt sich aus der Anzahl und Lage der verwendeten Teilungsebenen. Deren Ableitung aus den polygonalen Flächen des 3D-Gebäudemodells ist auf verschiedene Art und Weise möglich. Die dargestellte Vorgehensweise stellt daher nur eine von vielen Möglichkeiten dar. Beispielsweise ist auch die Anwendung der Hough-Transformation oder Cluster-Verfahren denkbar. Die Ergebnismenge sollte jedoch möglichst klein sein, und nur Ebenen enthalten, die, sofern sie sich nicht innerhalb des Minimalumschließenden Rechtecks schneiden, eine einstellbare Minimaldistanz zueinander entfernt sind. Durch diese Einschränkung wird die Erzeugung von Zellen mit zu schmalen Flächen weitestgehend verhindert. Somit ist die Minimaldistanz auch gleichzeitig der Generalisierungsschwellwert.

5.2.1 Selektion der Eingabepolygone

Zur Ableitung der Teilungsebenen für die 2.5D-Zellenzerlegung werden nur Polygone des Gebäudemodells verwendet, die nicht zum Dach gehören. Diese werden mithilfe des Normalenvektors identifiziert. Polygone mit einer vertikalen Komponente (z-Wert) größer Null sind Dachpolygone und werden weggelassen. Die Konstellation, dass zwei Dachpolygone übereinander liegen, kommt bei aus Luftdaten rekonstruierten 3D-Gebäudemodellen nicht vor und wird daher außer Acht gelassen.

5.2.2 Pufferbildung

Für jedes Polygon des Modells wird ein Puffer erzeugt der durch zwei parallele Ebenen begrenzt ist. Diese beiden Begrenzungsebenen sind anfangs identisch mit der Ebene in der das Polygon liegt. Damit durch die Zellenzerlegung eine vereinfachte Geometrie des 3D-Gebäudemodells entstehen kann, muss die Anzahl der Teilungsebenen, die identisch ist mit der Anzahl der Puffer, verringert werden. Dies geschieht indem nahezu parallele Puffer, die sich innerhalb einer gewissen Distanz zueinander befinden, zu einem breiteren Puffer zusammengefasst werden. Bei diesem Vorgang dürfen die beiden Begrenzungsebenen bis zu einer maximalen Generalisierungsdistanz verbreitert werden. Aus den Polygonen eines Puffers, die sich innerhalb des Bereichs zwischen den Begrenzungsebenen befinden, wird später die Teilungsebene berechnet. Dies ist natürlich nur für Polygone sinnvoll, die nahezu parallel zu den Begrenzungsebenen orientiert sind. Anders orientierte Polygone verfälschen die Teilungsebene und werden für die Berechnung der Ebenenparameter nicht mit berücksichtigt. Die Polygone eines Puffers werden daher in ebenenparallele und allgemeine Polygone unterschieden. Entitäten beider Polygonarten müssen vollständig im Pufferbereich enthalten sein. Die ebenenparallelen Polygone sind bis auf eine maximal erlaubte Winkelabweichung parallel zu den beiden Begrenzungsebenen des Puffers orientiert. Die allgemeinen Polygone dürfen beliebig orientiert sein. Dabei handelt es sich vor allem um kleine Seitenflächen von Vorsprüngen, Einbuchtungen, etc. Bei der Pufferbildung wird jedes Polygon genau einem Puffer zugeordnet und nicht allen Puffern, in dessen Bereich sich das Polygon befindet. Dadurch wird gewährleistet, dass kleine Polygonflächen keine eigenen Puffer erzeugen.

Zusammengefasst besteht jeder Puffer $B = \{E, F_E, F_A\}$ aus zwei Begrenzungsebenen und den Mengen der ebenenparallelen Polygone F_E und der allgemeinen Polygone F_A . Da die beiden Begrenzungsebenen parallel sind, reicht für die Definition des Pufferbereichs das Quintupel $E=(A, B, C, D_{min}, D_{max})$ aus. Die Definition der Begrenzungsebenen ist dann durch die beiden Gleichungen $Ax+By+Cz+D_{min}=0$ und $Ax+By+Cz+D_{max}=0$ gegeben.

Der Algorithmus zur Bildung der Puffer verwendet die folgenden Hilfsfunktionen, welche an dieser Stelle kurz erläutert werden sollen:

- **Fläche(B)**: Beim Vereinigen der Puffer sollen möglichst wenige Puffer entstehen, die jedoch sehr viele Polygone in ihrem Pufferbereich eingrenzen. Für einen qualitativen Vergleich ist daher eine Funktion $Fläche(B)$ definiert, welche für den Puffer B den Flächeninhalt der ebenenparallelen Polygone F_E aufsummiert.
- $B_{neu} = \mathbf{VereinigePuffer}(B_1, B_2)$ fasst die beiden Puffer B_1 und B_2 zu Puffer B_{neu} zusammen:
 1. Die Menge der ebenenparallelen Polygone F_E aus B_1 und B_2 , sowie die Menge der allgemeinen Polygone F_A aus B_1 und B_2 werden zu den neuen Mengen F_E und F_A vereinigt:

$$B_{neu} = \{E_{neu}, F_{1_E} \cup F_{2_E}, F_{1_A} \cup F_{2_A}\}$$

2. Die Normalenvektoren von E_{neu} der n ebenenparallelen Polygone F_E aus B_{neu} werden flächenanteilig zum neuen Normalenvektor $(A_{neu}, B_{neu}, C_{neu})$ gemittelt:

$$\vec{n}_{neu} = (A_{neu}, B_{neu}, C_{neu}) = \frac{\sum_{i=0}^n \text{Fläche}(F_{E_i}) \vec{n}_i}{\sum_{i=0}^n \text{Fläche}(F_{E_i})}$$

3. Die den Puffer begrenzenden Werte D_{min} und D_{max} werden durch Einsetzen aller Polygonpunkte V_i aus F_E und F_A in die Ebenengleichungen berechnet.

$$D_{min} = \min(-A_{neu} V_i x - B_{neu} V_i y - C_{neu} V_i z)$$

$$D_{max} = \max(-A_{neu} V_i x - B_{neu} V_i y - C_{neu} V_i z)$$

- **Vereinigbar**(B_1, B_2): Zwei Puffer B_1 und B_2 können zusammengefasst werden, wenn der Winkel zwischen den Normalenvektoren (A_1, B_1, C_1) und (A_2, B_2, C_2) geringer als der Winkelschwellwert ε_w ist und die Distanz zwischen den beiden neuen Begrenzungsebenen ($D_{max} - D_{min}$) kleiner dem Distanzschwellwert ε_D bleibt. Die Funktion **Vereinigbar**(B_1, B_2) liefert *wahr* zurück, wenn aus der Vereinigung der Puffer B_1 und B_2 ein neuer gültiger Puffer entstehen würde; andernfalls wird *falsch* zurückgegeben.

$$\varepsilon_w < \arccos \frac{(A_1 \cdot A_2 + B_1 \cdot B_2 + C_1 \cdot C_2)}{\sqrt{A_1^2 + B_1^2 + C_1^2} \cdot \sqrt{A_2^2 + B_2^2 + C_2^2}}$$

$$\varepsilon_D < \text{abs}(D_{max} - D_{min})$$

- $B_{neu} = \mathbf{Einschließen}(B_1, B_2)$ fügt Puffer B_2 dem Puffer B_1 hinzu:
 1. Die Menge der ebenenparallelen Polygone F_E und der allgemeinen Polygone F_A aus B_2 werden der Menge der allgemeinen Polygone F_A von B_1 hinzugefügt. Die Menge der ebenenparallelen Polygone F_E von B_1 bleibt unverändert. Dadurch wird sichergestellt, dass sich die Orientierung des Puffers B_1 durch das Hinzufügen des Puffers B_2 nicht ändert.

$$B_{neu} = \{E_1, F_{1_E}, F_{1_A} \cup F_{2_E} \cup F_{2_A}\}$$

2. Die den Puffer begrenzenden Werte D_{min} und D_{max} werden durch Einsetzen aller Polygonpunkte V_i aus F_E und F_A in die Ebenengleichungen berechnet.

$$D_{\min} = \min(-AV_i x - BV_i y - CV_i z)$$

$$D_{\max} = \max(-AV_i x - BV_i y - CV_i z)$$

- **Einschließbar**(B_1, B_2): Puffer B_2 kann dem Puffer B_1 hinzugefügt werden, wenn dadurch die Distanz zwischen den Pufferbegrenzenden Ebenen kleiner dem Distanzschwellwert ε_D bleibt.

$$\varepsilon_D < \text{abs}(D_{\max} - D_{\min})$$

Der Algorithmus startet mit einer Liste bestehend aus n Puffern. Jeder Puffer wurde aus genau einem Polygon gebildet. Die Ableitung der Puffer bzw. Teilungsebene erfolgt iterativ, wobei in jeder Iteration der vom Flächeninhalt der beinhalteten Polygone größte und damit wichtigste Puffer gefunden wird. Jede Iteration folgt dem folgenden Ablauf:

1. Sortiere die Pufferliste nach dem Flächeninhalt der Puffer
2. Vergleiche jeden Puffer B_1 in der Liste mit den nachfolgenden Puffern B_2 . Wenn *Vereinigbar*(B_1, B_2) den Wert *wahr* zurückliefert, so vereinige die Puffer mittels *Vereinige*(B_1, B_2) und füge den Ergebnispufer der Liste hinzu. Puffer B_1 und B_2 werden aus der Liste entfernt.
3. Vergleiche jeden Puffer B_1 in der Liste mit den nachfolgenden Puffern B_2 . Wenn *Einschließbar*(B_1, B_2) den Wert *wahr* zurückliefert, so vereinige die Puffer mittels *Einschließen*(B_1, B_2) und füge den Ergebnispufer der Liste hinzu. Puffer B_1 und B_2 werden aus der Liste entfernt.
4. Teste für alle noch nicht im größten Puffer befindlichen Polygone, ob diese sich vollständig in dessen Pufferbereich befinden und füge sie gegebenenfalls dem Puffer hinzu. Die Polygone werden aus den eigenen Puffern entfernt, der wiederum gelöscht werden wenn sich keine weiteren Polygone in ihrem Bereich befinden.
5. Berechne aus den ebenenparallelen Polygonen flächenanteilig den gemittelten Parameterwert D der Ebenenformel $Ax+By+Cz+D=0$. Die Werte A, B, C und D sind die Parameter der gefundenen Teilungsebene.
6. Entferne alle Polygone im Puffer selbst sowie alle Polygone, die sich vollständig innerhalb des Bereichs +/- dem halben Distanzschwellwerts ε_D von der gefundenen Ebene befinden aus der nachfolgenden Verarbeitung.

Da der Grundriss für die Charakteristik des Gebäudes sehr wichtig ist, verläuft der oben dargestellte Algorithmus in zwei Phasen. In der ersten Phase werden nur Puffer aus Polygonen, die an den Grundriss grenzen vergrößert. Ergeben sich dabei keine Änderungen mehr, d.h. es werden keine neuen Puffer gefunden, werden in der nächsten Phase Puffer aus allen verbleibenden Polygonen gebildet. Durch diese Vorgehensweise wird eine gute Approximation des Gebäudegrundrisses bevorzugt. Die aus dem Gebäudeinneren entstandenen Teilungsebenen verlaufen entlang großer Höhenunterschiede in der Dachstruktur.

Das Ergebnis ist eine Menge von Teilungsebenen. Jedoch sind nicht alle gefundenen Ebenen für die nachfolgenden Schritte relevant. Vor allem Teilungsebenen, die aus sehr wenigen und kleinen Polygonen abgeleitet wurden, stören die nachfolgende Prozessierung. Es ist daher notwendig, alle Ebenen, welche aus Puffern mit einem Flächeninhalt unter einem gesetzten Schwellwert entstanden sind, aus der Ergebnismenge zu entfernen. Um die Gesamtzahl der verwendeten Schwellwerte nicht unnötig zu vergrößern, wird das Quadrat des Distanzschwellwerts ε_D verwendet. Das lässt sich darin begründen, dass, wenn der Flächeninhalt der Pufferpolygone kleiner dem Quadrat ist, die größte Kante auch kleiner als der Distanzschwellwert ist und damit die Polygone aus dem Modell entfernt werden können.

Verwendete Schwellwerte

Bisher wurden zwei Schwellwerte eingeführt: der Winkelschwellwert ε_w und der Distanz- oder Pufferschwellwert ε_D für die Fassadenpolygone. Die Teilungsebenen werden ausschließlich über diese beiden Schwellwerte gefunden. Aus Tests hat sich erfahrungsgemäß ein Distanzschwellwert von circa sieben bis zehn Metern bewährt. Für Gebäude mit geringer Grundfläche können auch kleinere Schwellwerte verwendet werden. Für komplexe Objekte kann ein zu kleiner Schwellwert jedoch zu sehr vielen Zellprimitiven und dadurch wieder zu einem übermäßigen Detailreichtum führen. Wird der Schwellwert größer als die Distanz von einer Seite des Gebäudes zur anderen Seite verwendet, so werden die minimal benötigten Teilungsebenen nicht gefunden. Das bedeutet, dass durch die Zellenzerlegung mittels der Teilungsebenen kein Gebäudeumriss entsteht. Stattdessen entstehen teilweise unendlich große Zellen, die später komplett entfernt werden, was einen Wegfall von Gebäudeteilen in der generalisierten Form oder dem gesamten Gebäude zur Folge hat.

Dem Winkelschwellwert wurde in der Implementierung der feste Wert von 10° gegeben. Empirische Untersuchungen ergaben keine Verbesserung der beobachteten Ergebnisse bei der Verwendung eines größeren Winkelschwellwertes. Dies begründet sich in der Tatsache, dass die Inklusion eines Polygons in einen Puffer abhängig vom Distanzschwellwert ist. Durch den Winkelschwellwert wird nur die Zuordnung zur Menge der ebenenparallelen oder allgemeinen Polygone geregelt. Der Winkelschwellwert ist daher nur für Polygonflächen mit einer kleinen Ausdehnung relevant. Sofern ein großflächiges Polygon nicht parallel zu einer anderen Fläche ist, erzeugt es unabhängig vom Winkelschwellwert auch immer einen eigenen Puffer.

5.2.4 Ausgleichung der Teilungsebenen

Da die Parameter der Teilungsebenen unabhängig von den anderen Teilungsebenen berechnet werden, können geometrischen Relationen zwischen den Ebenen nicht garantiert werden. Die Ebenen bestimmen jedoch unmittelbar die Form des generalisierten Gebäudes. Daher ist es notwendig, die gewünschten geometrischen Relationen wie Parallelität und Rechtwinkligkeit zwischen den Teilungsebenen an dieser Stelle zu erzwingen. Da koplanare Teilungsebenen durch die Pufferbildung nicht vorkommen können, muss diese geometrische Relation auch nicht explizit berücksichtigt werden. Zunächst werden alle Teilungsebenen strikt vertikal ausgerichtet, so dass keine schrägen Fassadenwände entstehen können. Anschließend wird die Parallelität und Rechtwinkligkeit der Teilungsebenen erzwungen. Dies erfolgt über den paarweisen Vergleich der Normalenvektoren. Ist der Winkel zwischen den Ebenen nahe 0° bzw. 90° , so wird die Teilungsebene mit dem kleineren Flächeninhalt $Fläche(B)$ an die Teilungsebene mit dem größeren Flächeninhalt angepasst. Dies geschieht über eine Drehung der Ebene um den Schwerpunkt, der sich aus den bei der Entstehung der Ebene im Pufferbereich befindlichen Polygonen berechnet. Die Drehachse und der Drehwinkel im Bogenmaß können über das Kreuzprodukt und dem Arkuskosinus des Punktprodukts der beiden Normalenvektoren berechnet werden.

5.3 Erzeugung der 2.5D-Zellenzerlegung

Die Zellenzerlegung des zu generalisierenden Gebäudemodells erfolgt durch die Teilung eines nach unten und oben begrenzten, unendlichen Raums entlang der gefundenen Teilungsebenen. Die vertikale Raumbegrenzung ist abhängig von dem gewünschten Ergebnis. Nach unten ist die Begrenzung immer durch den niedrigsten Gebäudepunkt fest vorgegeben, da ansonsten für Modelle mit stark abgechrägter Grundfläche und tief hängenden Dächern negative Zellhöhen entstehen können. Außerdem wird so gewährleistet, dass bei der Integration mit einem DHM das fertige Modell über die gesamte Grundfläche im Boden steckt und nicht etwa darüber schwebt. Die Gebäudegrundhöhe kann nach der Generalisierung auch noch durch ein Kappen des Bodens angepasst werden. Die obere vertikale Begrenzung ist abhängig vom gewünschten Ergebnis. Soll ein Blockgebäude erzeugt werden, so kann einfach die flächenabhängig gemittelte Höhe der Dachpolygone verwendet werden. Für die nachfolgende Konstruktion der generalisierten Dachstruktur wird ein gewisser Spielraum nach oben benötigt, so dass mindestens die doppelte Höhe der vertikalen Ausdehnung des Gebäudemodells als obere Begrenzung notwendig ist. So wird garantiert, dass sich die Teilungsebenen innerhalb des Anfangsblocks schneiden und dadurch wohlgeformte Zellen entstehen. Andernfalls kann eine zu niedrige Begrenzung zu einer Kappung der Dachfirste führen. Da ein unendlicher Raum in der Praxis ungeschickt zu verarbeiten ist, wird der Anfangsraum auch in der horizontalen Richtung begrenzt. Damit alle Schnitte sicher

innerhalb des Raums erfolgen und die äußeren Zellen sehr viel Leerraum enthalten, wird horizontal die doppelte Ausdehnung des Minimalumschließenden Rechtecks verwendet. Der Leerraum wird später benötigt, um die nicht zum Gebäudemodell gehörenden Randzellen sicher zu identifizieren und zu entfernen.

Die Zellenzerlegung erfolgt durch die wiederholte Teilung des Anfangsraums entlang aller Teilungsebenen. Im Vergleich zur Binären Raumteilung entsteht dabei jedoch keine hierarchische Struktur. Dies hat eine größere Anzahl von Zellen zum Ergebnis. Während beispielsweise eine einfache Zellenzerlegung des Grundrisses in Abbildung 5-5 aus fünf Gebäudezellen besteht, benötigt eine hierarchische Zerlegung für denselben Grundriss nur drei Zellen. Die Zellen der einfachen Zellenzerlegung sind jedoch regelmäßig geformt und spiegeln die Symmetrien des Gebäudes besser wieder. Die Konstruktion der vereinfachten Dachform wird dadurch stark vereinfacht. Während für die Konstruktion der Dachform die Symmetrieeigenschaften der fünf Zellen genutzt werden kann, ist dies für die drei hierarchisch erzeugten Zellen nicht möglich.

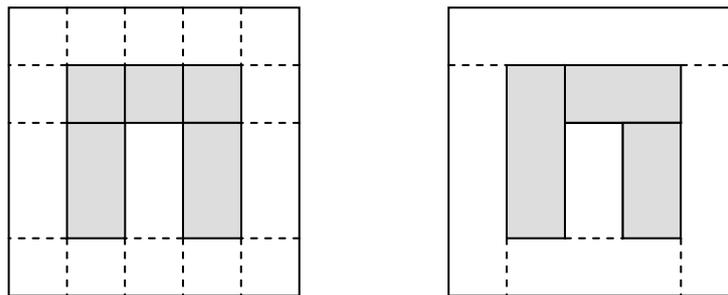


Abbildung 5-5: Eine einfache (links) und hierarchische (rechts) Zellenzerlegung des Gebäudegrundrisses.

5.4 Identifikation der Gebäudezellen

Aus der Zellenzerlegung ist die Teilmenge zu identifizieren, welche „zusammengeklebt“ das vereinfachte Gebäudemodell ergibt. Dafür wird die zweidimensionale Überdeckung der Zellen mit dem Grundriss berechnet. Hat eine Zelle eine hohe Überdeckung mit dem Grundrisspolygon, d.h. wird ein hoher Prozentsatz der Zellgrundfläche durch das Grundrisspolygon überdeckt, so wird die Zelle als Gebäudezelle klassifiziert. Ansonsten ist sie keine Gebäudezelle und wird aus der weiteren Verarbeitung entfernt. Vor allem die Randzellen haben sehr große Leerflächen. Dadurch ergibt sich eine geringe Überdeckung und sie können zweifelsfrei als Nichtgebäudezellen erkannt werden. Für alle anderen Zellen muss wiederum mittels eines Schwellwerts definiert werden, ab welcher prozentualen Überdeckung es sich um eine Gebäudezelle handelt. Um das fertige Gebäudemodell nur aus gut ausgefüllten Zellen zu generieren, wurde ein sehr hoher Überdeckungsschwellwert von 70% gewählt. Andere Schwellwerte müssen nicht zwangsläufig zu unterschiedlichen Ergebnissen führen. Von höheren Schwellwerten ist jedoch abzuraten, da Gebäudezellen am Rand fast immer einen kleinen Leerbereich besitzen, wo die Fassade generalisiert wurde. Diese Zellen würden ansonsten wegfallen. Um dies zu verhindern, kann anschließend mittels der Überdeckung von Originalgrundriss und allen Gebäudezellen die Diskrepanz zwischen den Modellen berechnet werden. Ist diese zu groß, so kann der Schwellwert verringert werden, bis ein akzeptables Ergebnis entsteht.

Die Überdeckung kann mittels Flächenverschnidung berechnet werden. Ist F_1 die Grundfläche der Zelle und F_2 das Grundrisspolygon, so ist die prozentuale Überdeckung der Flächeninhalt der Schnittfläche $F_1 \cap F_2$ geteilt durch den Flächeninhalt von F_1 . Für eine effiziente Berechnung der Schnittfläche kann beispielsweise auf den Algorithmus von Vatti [1992] zurückgegriffen werden. Eine weitere Möglichkeit ist die Diskretisierung der zu untersuchenden Zelle in gleichgroße Rasterflächen (siehe Abbildung 5-6). Mittels Punkt-in-Polygon-Test werden zuerst die Rasterflächen identifiziert, deren Mittelpunkte sich innerhalb der Zelle befinden. Die Ergebnisrasterflächen werden anschließend mit derselben Methode gegen den Grundriss getestet. Dadurch ergibt sich eine Unterscheidung in Grundriss- und Nichtgrundrissraster, aus deren Verhältnis zur Gesamtzahl der Zellraster ein Näherungswert für die Überdeckung berechnet werden kann. Im Gegensatz zur genauen Berechnung der Überdeckung erlaubt dieser Ansatz eine einfache Implementierung. Die Genauigkeit und die sich daraus ergebende Rechenzeit lässt sich über die Rastergröße steuern. Da der Überdeckungsschwellwert keiner wissenschaftlichen Grundlage gebiert, reicht diese Näherungslösung gänzlich aus.

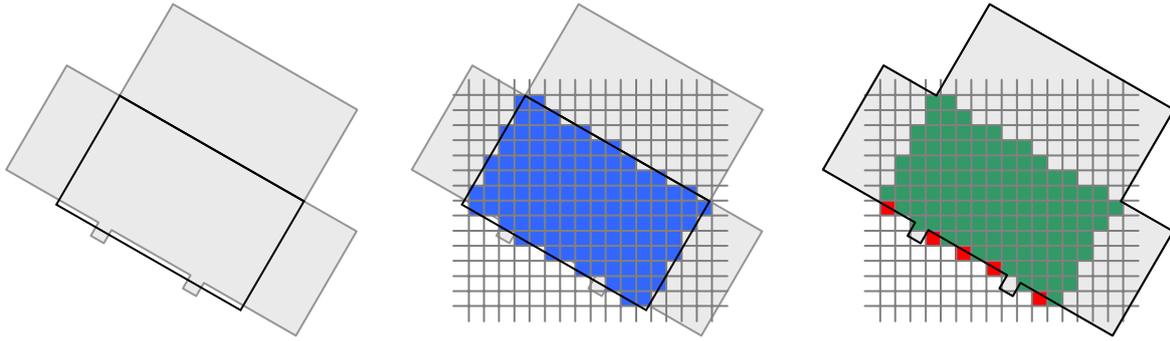


Abbildung 5-6: Gebäudegrundriss mit überlagerter Zelle (links), blauer Untersuchungsbereich nach Discretisierung (Mitte) und daraus resultierende Unterscheidung in grüne Grundriss- und rote Nichtgrundrissraster (rechts).

Blockgebäude

Um aus den Gebäudezellen ein Blockgebäude zu erhalten, müssen die Zellen nur noch „zusammengeklebt“ werden. Dabei handelt es sich um eine eingeschränkte Form der Vereinigung von Festkörpern, bei dem die Objekte sich nicht überschneiden dürfen. Durch die Art und Weise wie die Zellen mittels Zerschneidung entstanden sind teilen sich angrenzende Zellen immer eine gemeinsame Fläche. Beim Zusammenfügen zweier Zellen werden diese Grenzflächen, sowie gegebenenfalls deren Kanten, einfach aus beiden Objekten entfernt.

5.5 Generierung der 3D-Dachform

Soll kein Blockgebäude erzeugt werden, sondern ein Gebäude mit einer echten Dachstruktur, so wird mit den Blockprimitiven weitergearbeitet. Dabei sollten die Blockprimitive wie schon beschrieben, die doppelte oder dreifache Gebäudehöhe besitzen. Für jedes Blockprimitiv wird jetzt einzeln eine Dachstruktur erzeugt. Die verwendete Methode ist weitestgehend identisch mit der bisherigen Vorgehensweise zur Erzeugung der Blockprimitive. Die daraus resultierenden 3D-Primitive werden abschließend genau wie beim Blockgebäude zusammengeklebt.

Der erste Schritt ist daher wieder die Ableitung von Teilungsebenen. Hierfür werden die Puffer jetzt nicht aus den Fassadenpolygone sondern aus den Dachpolygone generiert. Obwohl die Verarbeitung anschließend per Primitiv weitergeführt wird, geschieht dies global für das gesamte Gebäude. Ansonsten würden die erzeugten Dachflächen an den Übergängen zwischen den Primitiven Höhen- und Neigungsunterschiede aufweisen. Weiterhin ist zu beachten, dass der Pufferschwellwert ε_D nicht so groß wie für die Fassadenpuffer gewählt werden darf. Ansonsten landen für sehr flache Dachstrukturen alle Dachpolygone in demselben Puffer und es entstehen Primitive mit Pultdach. Ein Pufferschwellwert zwischen drei und fünf Metern hat sich in experimentellen Vergleichen bewährt.

Die global gefundenen Teilungsebenen für die Dächer werden anschließend nicht auf alle Blockprimitive angewendet. Sondern ein Primitiv wird nur durch diejenigen Teilungsebenen zerschnitten, deren definierenden Polygone eine minimale Überdeckung des Primitivgrundrisses aufweisen. Damit werden unnötige Schnitte verhindert und die Anzahl der entstehenden 3D Primitive minimiert.

Im Gegensatz zu den Teilungsebenen aus Fassadenpolygone können die Teilungsebenen aus Dachpolygone auch horizontal oder schräg horizontal die Primitive zerschneiden. Das bedeutet aber, dass 3D Primitive auch übereinander angeordnet sein können und die Unterscheidung in Gebäude- und Nichtgebäudeprimitive nicht mehr über den Grundriss berechnet werden kann. Die Überdeckung muss jetzt im dreidimensionalen Raum mittels Verschneidung von Körpern gelöst werden. Ähnlich der Definition im vorhergehenden Abschnitt gilt, dass wenn P_1 das untersuchende Primitiv und P_2 das Originalmodell ist, so ist die prozentuale Überdeckung der Rauminhalt der Verschneidung $P_1 \cap P_2$ geteilt durch den Rauminhalt von P_1 .

Für die Berechnung der Rauminhalte sollen an dieser Stelle zwei Verfahren vorgestellt werden. Das erste Verfahren liefert sehr genaue Ergebnisse, ist jedoch recht aufwendig zu implementieren. Das zweite Verfahren liefert nur Näherungswerte, ist dafür einfacher zu implementieren und kann auch mit fehlerhaften Datensätzen robust umgehen.

Die Oberfläche des Polyeders muss für das erste Verfahren geschlossen sein, aus Dreiecken bestehen und die Eckpunkte aller Dreiecke von Außen betrachtet gegen den Uhrzeigersinn orientiert sein. Sei $Raum(t, p)$ der mit Vorzeichen behaftete Rauminhalt des Tetraeders der durch den Punkt p und das Dreieck t gegeben ist. Dieser lässt sich wie in [O'Rourke 1998] beschrieben mit einer 4×4 Determinante berechnen. Wird Punkt p beliebig gewählt, so berechnet sich der Rauminhalt des Polyeders V über die Summe $Raum(t_i, p)$ der durch die n Dreiecke t_i gegebenen Tetraeder.

Die Formel lässt sich ähnlich begründen wie die Gaußsche Flächenformel zur Berechnung des Flächeninhalts eines beliebigen Polygons. Die dem Punkt p abgewandten Dreiecke ergeben positive Rauminhalte, die jedoch auch den Raum zwischen Objekt und Punkt mit beinhalten. Dieser Raum wird durch die negativen Rauminhalte der dem Punkt p zugewandten Dreiecke wieder abgezogen.

Der Rauminhalt des zu untersuchenden Primitivs P_1 lässt sich, wie oben beschrieben, nach der Triangulierung der Oberfläche direkt berechnen. Für die Berechnung des Rauminhalts des Originalmodells P_2 innerhalb des Primitivs P_1 muss das Originalmodell jedoch zunächst mit den Primitivseiten von P_1 geschnitten werden. Das Ergebnis spiegelt jedoch sehr genau die gewünschte prozentuale Überdeckung wieder.

Im zweiten Algorithmus wird ähnlich wie beim Grundrissverfahren. Zuerst wird der axial ausgerichtete Hüllquader (Axis-Aligned Bounding Box) des zu untersuchenden Primitivs P_1 berechnet. Dieser Quader wird in ein gleichmäßiges, dreidimensionales Gitter mit einer festen Anzahl von Zellen unterteilt. Das Ergebnis ist genauer je feiner diese Unterteilung erfolgt. Nun wird mittels Punkt-im-Polyeder-Test ermittelt, welche Zellen sich innerhalb des Primitivs P_1 befinden. Die Zellen außerhalb des Primitivs werden nicht weiter betrachtet. Für die verbleibenden Zellen wird weiter getestet, ob sie sich auch innerhalb des Originalmodells befinden. Hierfür wird ähnlich dem Punkt-im-Polygon-Test nach Jordan die Anzahl der Schnittpunkte eines Strahls mit den Polygonen des Polyeders gezählt. Bei einer ungeraden Zahl befindet sich der Punkt innerhalb des Polyeders, ansonsten außerhalb. Die prozentuale Überdeckung ist dann das Verhältnis der Anzahl der Punkte innerhalb des Ursprungsmodells zur Anzahl der Punkte innerhalb des Primitivs.

Das Ergebnis einer solchen Zellerlegung ist in Abbildung 5-7 dargestellt. Nach der Identifikation der 3D-Gebäudeprimitive werden diese zum Ergebnismodell zusammengeklebt. Vorher können jedoch die nachfolgend beschriebenen Optimierungsverfahren angewandt werden.

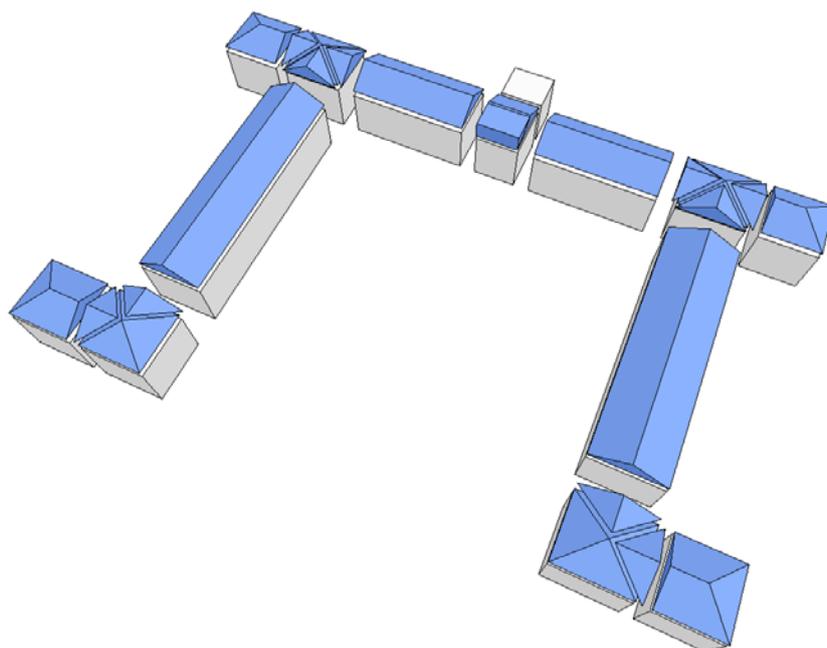


Abbildung 5-7: Zellerlegung der generalisierten Dachstruktur des Neuen Schlosses von Stuttgart.

5.6 Optimierung der Primitivauswahl

Durch die Hinzunahme der Dachteilungsebenen, die im Allgemeinen recht uneingeschränkt im 3D-Raum orientiert sein können, entstehen durch die Verschneidung oftmals auch sehr kleine Primitive. Für diese Primitive ist die Unterscheidung in Gebäude- und Nichtgebäudeprimitive nach dem oben beschriebenen Überdeckungskriterium nicht immer ausreichend. Um eine möglichst einfache Gebäudestruktur zu erhalten, ist die Auswahl der Primitive, die zum endgültigen 3D-Gebäudemodell zusammengefügt werden, weiter zu optimieren. Zu diesem Zweck wurden heuristische Regeln zum Füllen von Löchern und zum Entfernen von unvorteilhaften Primitiven aufgestellt. Der Optimierungsprozess wird anschließend durch eine globale Lageverbesserung der Primitivecke abgeschlossen.

5.6.1 Auffüllen von Löchern

Bei Gebäudemodellen mit sehr komplexer Dachstruktur kann es vorkommen, dass durch das Entfernen der korrekt identifizierten Nichtgebäudeprimitive Löcher im generalisierten Modelldach entstehen. Diese Löcher können oftmals sehr einfach geschlossen werden indem diese Primitive wieder dem Modell zugefügt werden. Die entsprechenden Primitive können mittels folgender Regel identifiziert werden:

1. Für jedes als nicht zum Gebäude zugehörig erkannte Primitiv wird getestet, ob die Gesamtfläche der Polygone, welche an schon akzeptierte Gebäudeprimitive grenzen, größer ist als die Fläche der Polygone, die an Nichtgebäudeprimitive grenzen. Ist dies der Fall, so schließt das zu testende Primitiv ein Loch und sollte folglich im Modell beibehalten werden.

Durch die mehrfache Anwendung dieser recht einfachen Methode zum Auffüllen von Löchern werden Gebäudemodelle mit allgemein einfacherer Dachstruktur erzeugt (siehe Abbildung 5-8).

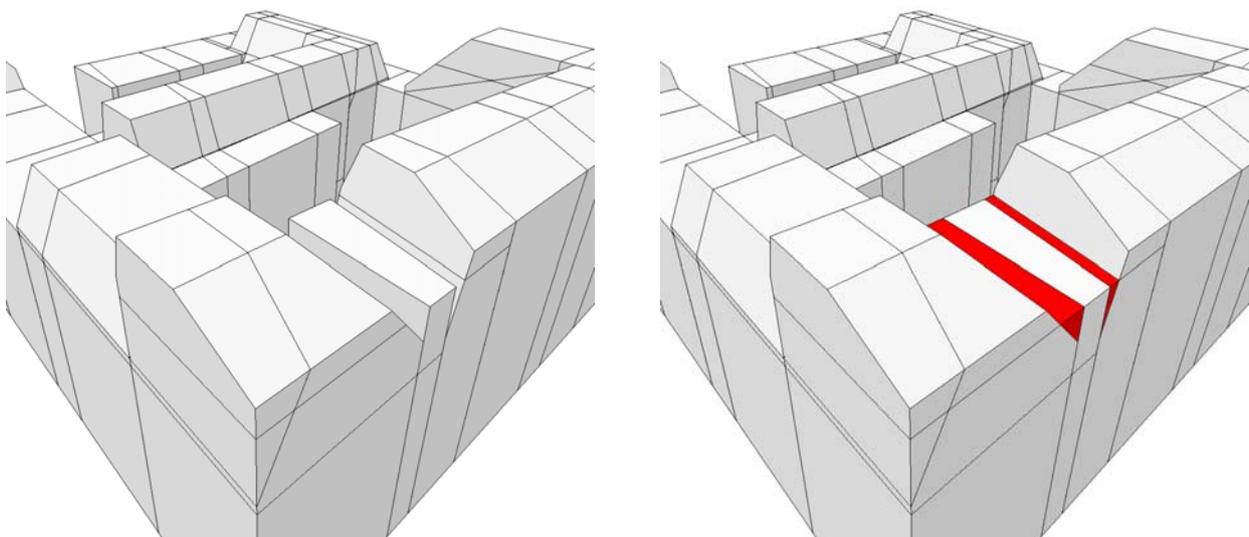


Abbildung 5-8: Zellerzerlegung eines generalisierten Gebäudemodells vor (links) und nach (rechts) der Auffüllung von kleinen Löchern.

5.6.2 Entfernen von Primitiven

Recht ähnlich verhält es sich mit Primitiven, die zwar korrekt als Gebäudeprimitive erkannt wurden, aber nur störende Details zur Dachstruktur hinzufügen. Viele dieser Primitive zeichnen sich dadurch aus, dass nur eine Fläche an ein Nachbarprimitiv grenzt. Mit den folgenden zwei Regeln wird entschieden, ob Primitive weggelassen werden können (vgl. Abbildung 5-9):

1. Wenn die Grenzfläche des Primitivs nicht waagrecht ist, aber eine andere Fläche des Primitives, so kann es entfernt werden. Damit wird verhindert, dass auf eine schräge Dachfläche, beispielsweise von einem Giebedach, noch Elemente draufgesetzt werden.
2. Zeigt der Normalenvektor einer Fläche, an der kein Primitiv grenzt, nach unten, so würde das Primitiv einen Überstand erzeugen und muss weggelassen werden.

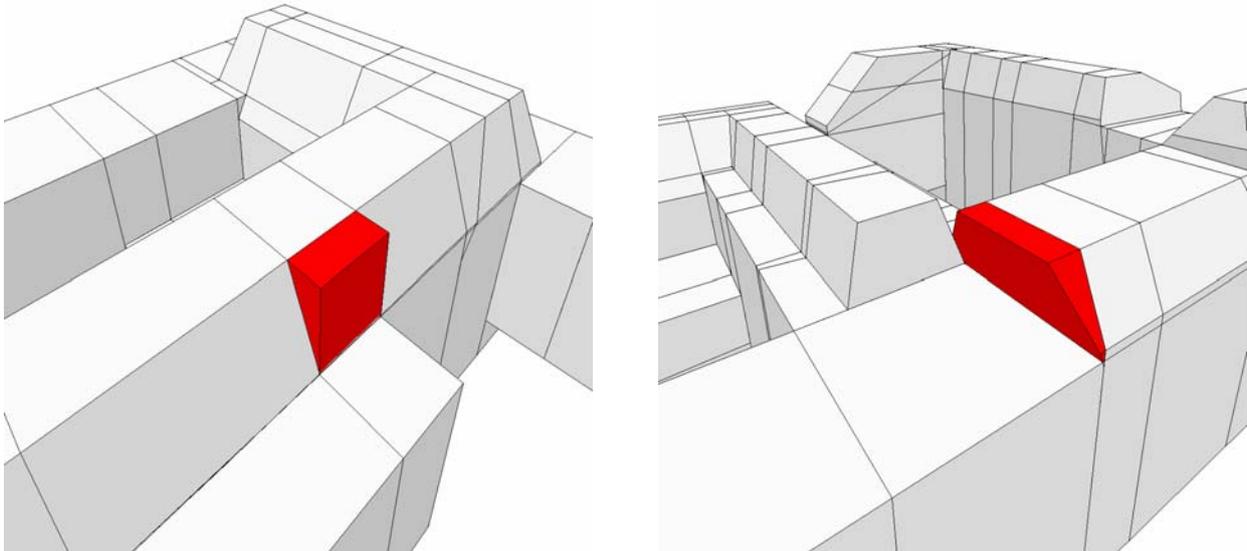


Abbildung 5-9: Zellerlegung eines generalisierten Gebäudemodells mit zu entfernenden, roten Zelle.

5.6.3 Zusammenziehen von Knoten

Mittels der heuristischen Regeln kann nur die Auswahl der Primitive optimiert werden, welche zusammen das generalisierte Gebäudemodell ergeben sollen. Ungenauigkeiten, die zwischen angrenzenden Primitiven auftreten, können mit diesen Methoden jedoch nicht eliminiert werden. Stattdessen müssen die Gebäudeprimitive geometrisch verändert werden. Dafür wird ein Vertex-Clustering-Ansatz verwendet, welcher die Knoten, die nahe beieinander liegen, in einen gemeinsamen Punkt zusammenziehen. Dabei dürfen die Endpunkte nicht willkürlich gewählt werden sondern sie müssen eine Position besitzen, die keine Verzerrung des Gebäudes verursachen. Da die Knoten des generalisierten Grundrisses generell schon sehr gut liegen, werden deren x- und y-Positionen bei dem Verfahren bevorzugt. Diese müssen bei dem Vertex-Clustering auch nicht verändert werden. Vielmehr werden die Eckpunkte der generalisierten Dachform auf die x- und y-Position des nächstgelegenen Grundrisspunkts gezogen. Dadurch werden die Punkte der Dachtraufe am Grundriss ausgerichtet. Knoten dürfen jedoch nur verschoben werden, wenn die Distanz zu deren neuen Position geringer als ein vorher definierter Schwellwert ist. Um einen einheitlichen und zentrierten Dachfirst zu generieren, werden außerdem die x- und y-Koordinaten von den Mittelpunkten der Kanten und Grundflächen der Blockprimitive verwendet (siehe Abbildung 5-10). Knoten, die sich nicht in der Nähe eines der oben genannten Positionen befinden, werden an die x- und y-Position des nächstgelegenen Knoten verschoben, welcher den größten Flächeninhalt aller angrenzenden Polygone aufweist. Im nächsten Schritt werden die z-Positionen verändert. Hierfür wird aus den z-Positionen aller Eckpunkte mittels Clustering eine minimale Anzahl von Höhenwerten abgeleitet. Für einfache Dachstrukturen erhält man beispielsweise eine Grundriss-, Trauf- und Firsthöhe. Die Eckpunkte des generalisierten Gebäudemodells werden dann an die am nächsten gelegene Höhenstufe gezogen. Durch das Vertex-Clustering degenerieren Kanten, Polygone und Zellen die anschließend aus dem generalisierten Modell entfernt werden.

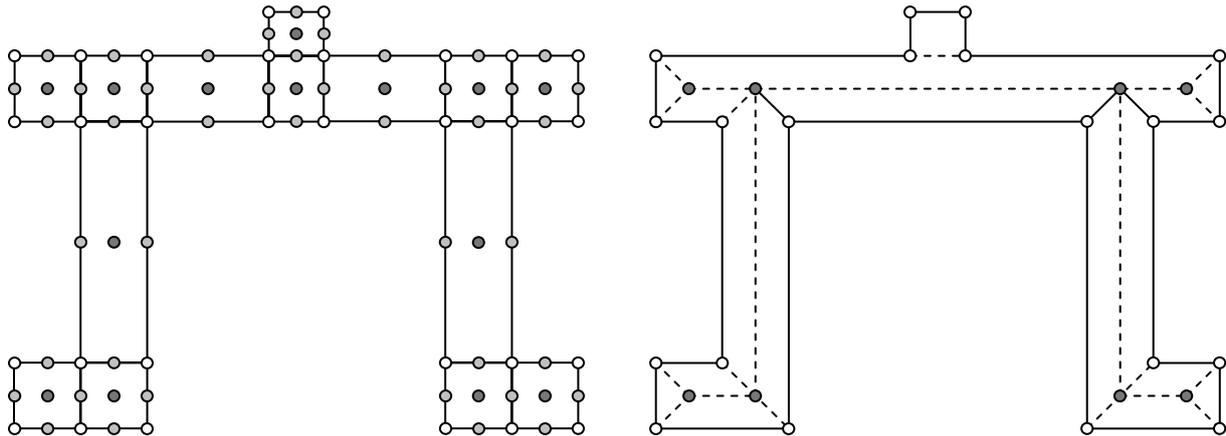


Abbildung 5-10: Zellenzerlegung des Neuen Schloss von Stuttgart mit allen Clusterzentren (links), sowie die resultierenden Dachflächen mit den tatsächlichen Clusterzentren (rechts).

5.7 Ergebnisse

Mit dem beschriebenen Generalisierungsalgorithmus wurden zahlreiche 3D-Gebäudemodelle vereinfacht. Die Ergebnisse sind exemplarisch in Abbildung 5-11 bis Abbildung 5-13 zu sehen. Bei der Auswahl der Modelle wurde vor allem darauf geachtet, eine Vielfalt unterschiedlichster Gebäudeformen darzubieten. Abbildung 5-11 zeigt die Ergebnisse anhand von vier manuell erzeugten Modellen. Diese besitzen sehr einfache Formen, welche jedoch die typischen Problemfälle für die Generalisierungsverfahren aus Kapitel 3 und Kapitel 4 darstellen. Zum Beispiel besitzen die zwei leicht verschobenen Doppelhaushälften keine identifizierbaren Merkmale und die Struktur ist für allgemeine Modelle nur schwer mittels expliziter Regeln zu beschreiben. Vielmehr entsteht die Komplexität des 3D-Modells durch die Verschiebung der ansonsten nicht vereinfachbaren Teilprimitive. Ein direkter Vergleich der originalen und generalisierten Modelle ist in der mittleren Spalte von Abbildung 5-11 gegeben. Die Flächen der vereinfachten, blauen Gebäudemodelle sind nicht immer identisch mit den Originalflächen. Dies kann auf die Schnittebenen zurückgeführt werden, welche die Menge der Originalflächen nicht exakt, sondern approximiert widerspiegeln. Besonders gut ist das an dem Modell der Doppelhaushälften zu beobachten. Das resultierende, lang gezogene Gebäudemodell mit dem einheitlichen Satteldach liegt genau zwischen den beiden Ausgangsmodellen, als ob die beiden Teilprimitive sich jeweils um die halbe Distanz zueinander hin bewegt hätten. Bei den Dachflächen des generalisierten Schlepddachs würde man jedoch erwarten, dass sie die Originalflächen exakt wiedergeben. Schließlich konnten die Schnittebenen, aus denen die Flächen entstanden sind, nur aus jeweils einer Fläche des Originalmodells gebildet werden. Durch das abschließende Zusammenziehen der Knoten sind jedoch die beiden Firstpunkte auf die Seitenmitte gezogen worden, so dass ein symmetrisches Satteldach entsteht. Vorbauten (Erker, Ausluchten, etc.) und Dachaufbauten (Gauben, Schornsteine, etc.), die kleiner als der Generalisierungsschwellwert sind, werden entfernt. Die originale Dachstruktur bleibt dabei unangetastet und wird bei dem Vorgang nicht zerstört.

Abbildung 5-12 bis Abbildung 5-13 zeigen die Ergebnisse anhand von 3D-Gebäudemodellen des Stuttgarter Stadtmodells, wie sie in Kapitel 2 und Wolf [1999] beschrieben sind. Die Ergebnisse werden für ausgewählte Modelle mit sowohl einfacher als auch komplexer Struktur gezeigt. Zu kleine Details im Grundriss und in der Dachstruktur werden entfernt, ohne die globale Gebäudeform zu verfälschen. So bleibt beispielsweise die charakteristische Form von Walm- und Mansarddächern in der Regel erhalten. Die von Staufenbiel [1973] benannten Strukturen wie Eck- und Seitenversprünge sowie Einsprünge sind, sofern diese geringer als der Generalisierungsschwellwert waren, im vereinfachten Modell nicht mehr vorhanden. Auch Häuserecken mit abgekappten Kanten werden wieder zu spitzen Ecken vervollständigt. Dabei bleiben die gebäudespezifischen, geometrischen Relationen wie Koplanarität, Parallelität und Rechtwinkligkeit der Grundrissflächen erhalten. Durch die Optimierung der First- und Giebelknoten werden außerdem die Dachflächen von Sattel-, Walm- und Mansarddächern symmetrisch ausgerichtet. Hervorzuheben sind noch die in Abbildung 5-13 zu sehenden 3D-Modelle von prominenten Gebäuden von Stuttgart. Sie weisen eine komplexe und detaillierte Grundriss- und Dachstruktur auf. Dabei konnte z.B. die Komplexität des Modells vom Neuen Schloss von Stuttgart von 2727 auf 82 Dreiecke reduziert werden. Auch die im Opernhaus und in der Kirche vorkom-

menden unterschiedlichen Dachformen werden vereinfacht, aber in Übereinstimmung mit dem Originalmodell korrekt wiedergegeben.

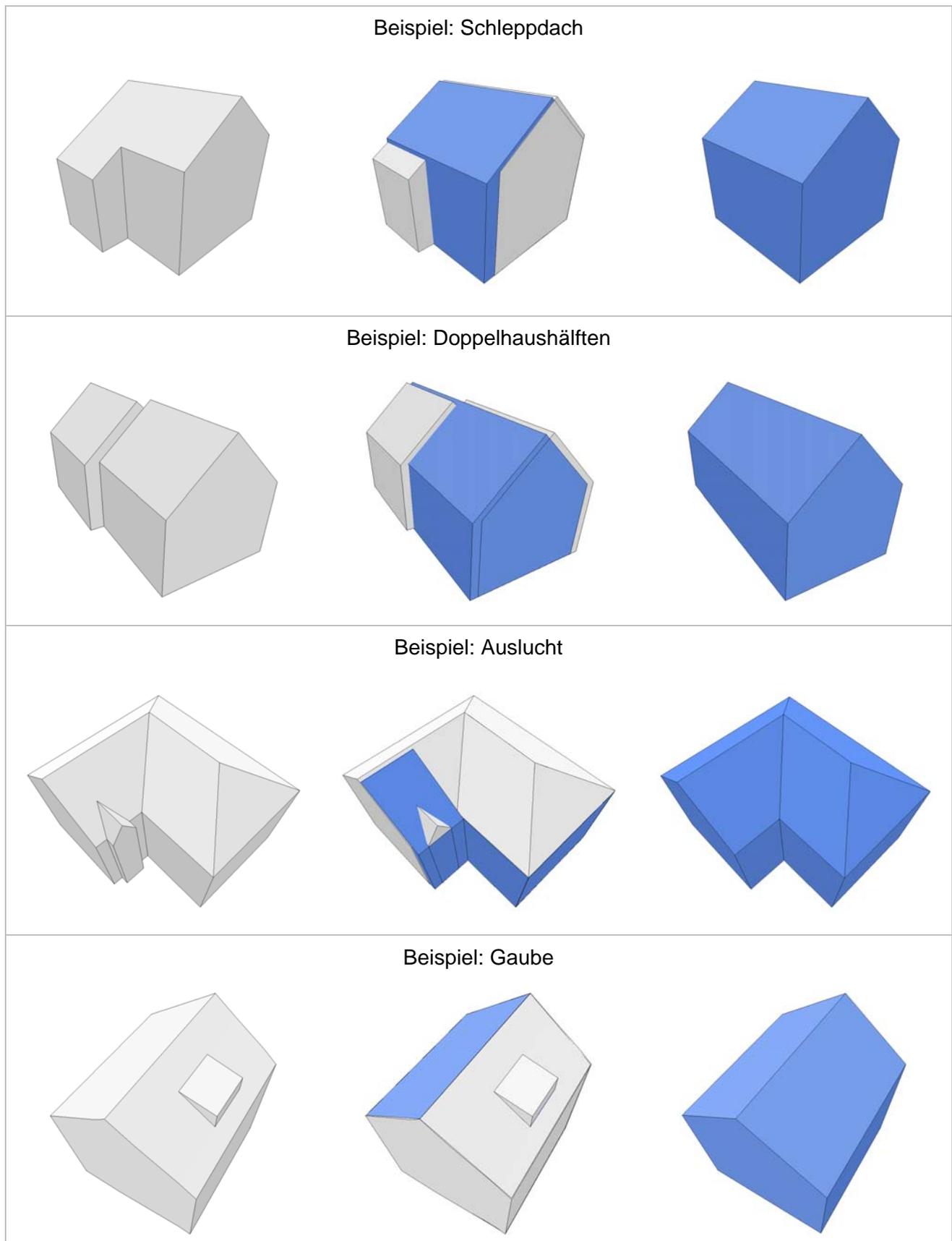


Abbildung 5-11: 3D-Gebäudemodelle im Original (links), generalisiert (rechts) und überlagert (Mitte).

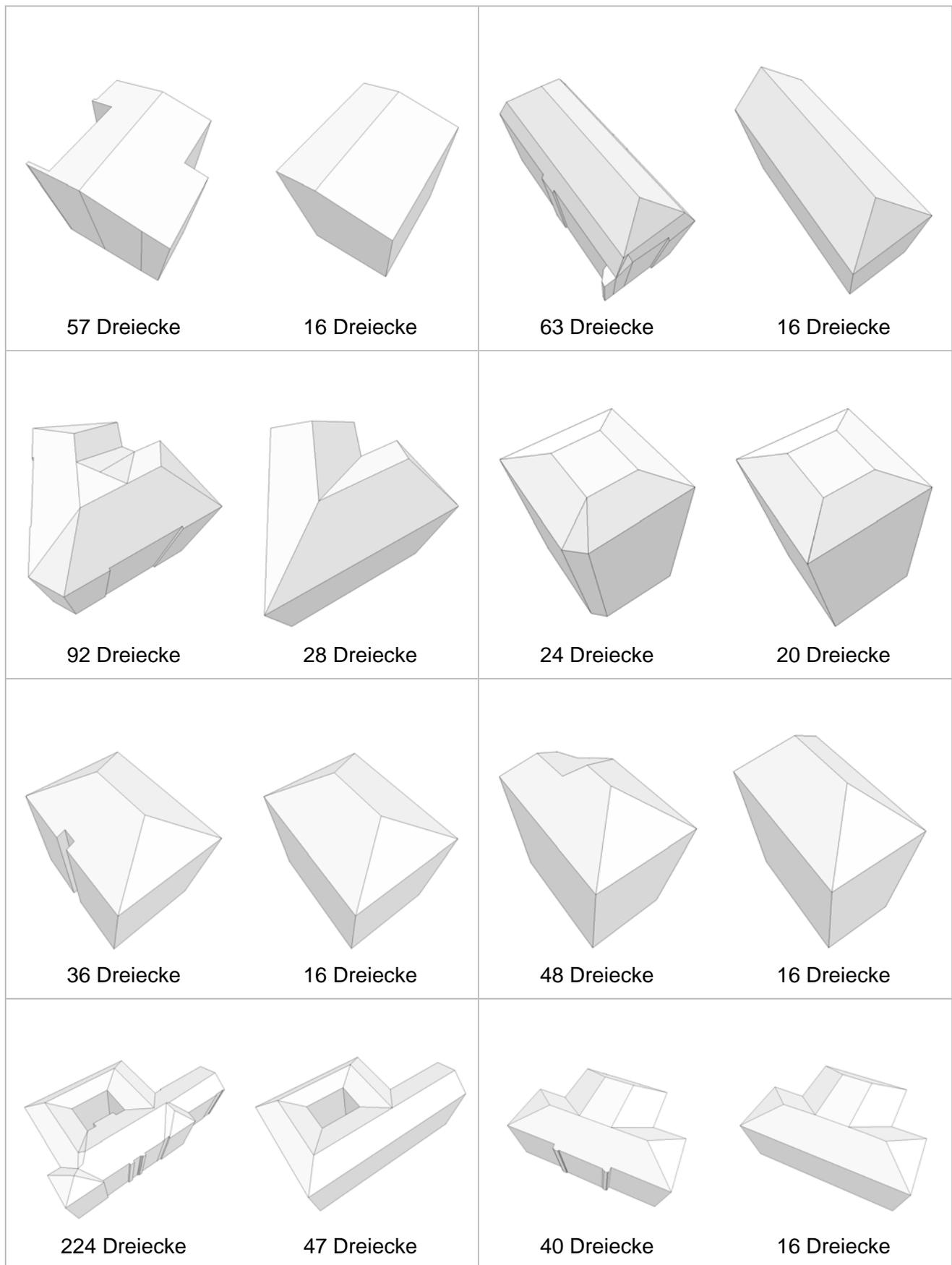


Abbildung 5-12: Gebäudemodelle im Original (links) und generalisiert (rechts).

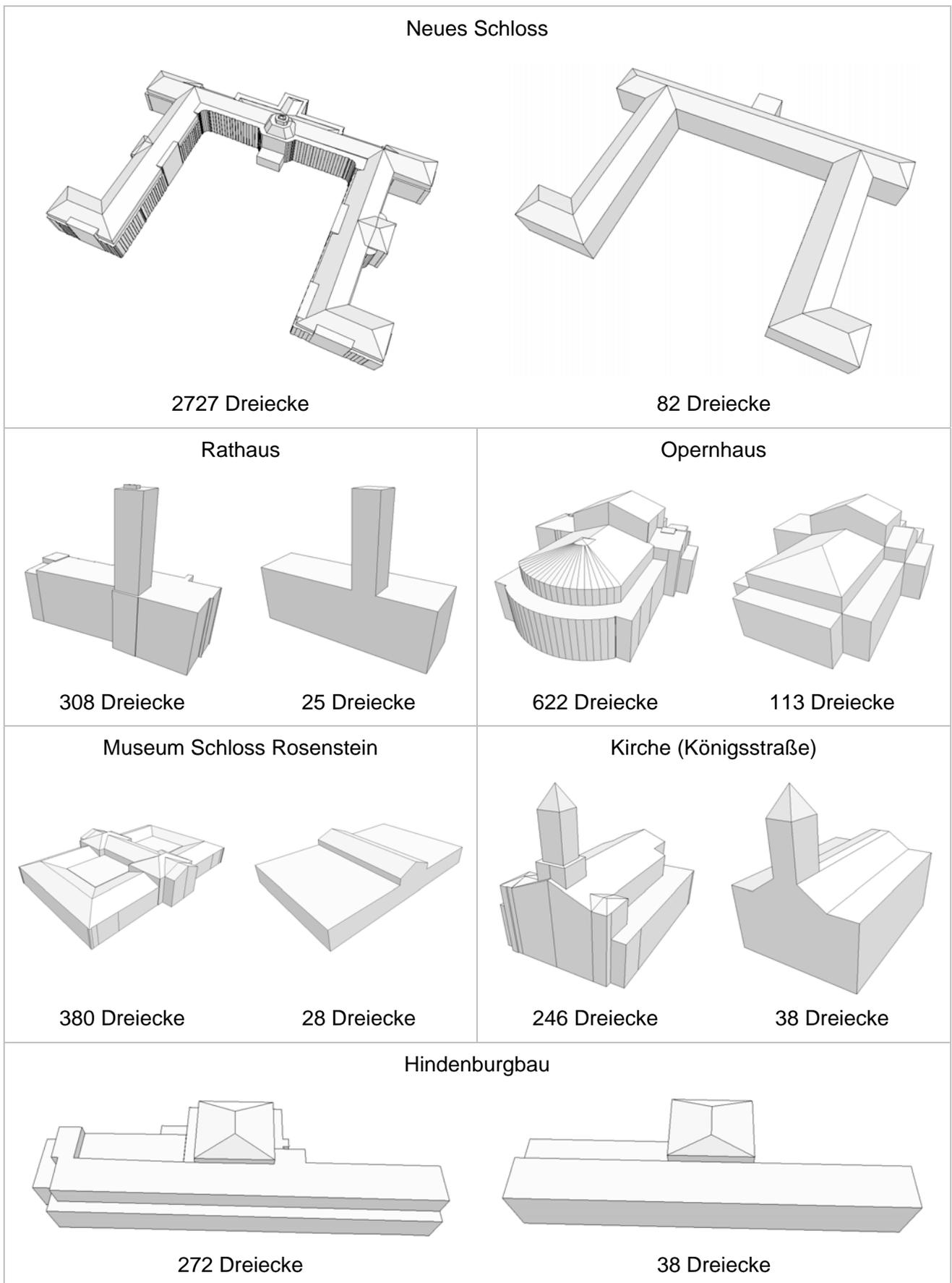


Abbildung 5-13: Gebäudemodelle im Original (links) und generalisiert (rechts).

5.8 Alternativer Ansatz zur Generierung der 3D-Dachform

Manchmal kann es vorkommen, dass trotz Anwendung der beschriebenen Optimierungsregeln, keine schönen Dachformen entstehen. Dies ist vor allem darin zu begründen, dass keine Interpretation der originalen Dachform vorgenommen wird. Vielmehr ist das Verfahren möglichst allgemein gehalten, um beliebige Dachgeometrien zu unterstützen. Dadurch können die folgenden Problemfälle auftreten:

1. Bei flachen Dachstrukturen lassen sich im Bereich des Blockprimitivs alle Dachpolygone einem einzigen Puffer zuordnen. Es wird folglich nur eine Teilungsebene generiert, aus der, unabhängig von der Ausgangsdachform, ein Pultdach erzeugt wird. Kann eine flache Dachform jedoch nicht erhalten werden, so wäre die Generalisierung zu einem Flachdach mittlerer Höhe sinnvoller (vgl. Abbildung 5-14).

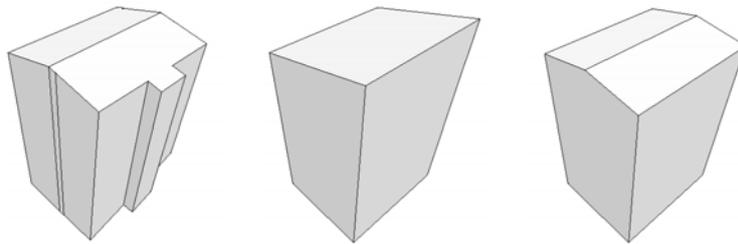


Abbildung 5-14: Original 3D-Gebäudemodell und Generalisierung zu einem Pult- und einem Satteldach.

2. Da die Dachneigungen der Teilungsebenen nicht zueinander ausgerichtet werden, entstehen beispielsweise bei der Generalisierung von Walmdächern häufig unsymmetrische Walme (vgl. Abbildung 5-15).

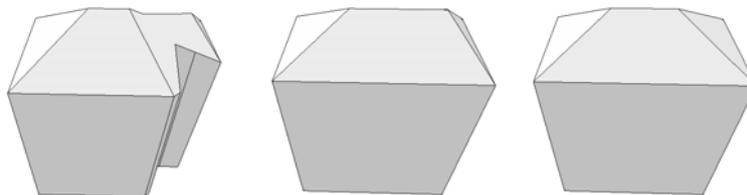


Abbildung 5-15: Original 3D-Gebäudemodell und Generalisierung zu einem unsymmetrischen und symmetrischen Walmdach.

3. Aufgrund von unterschiedlichen Traufhöhen erhalten manchmal Primitive nicht mehr die ausreichende Überdeckung mit dem Originalmodell. Sind diese zu groß, greifen auch die Optimierungsregel nicht mehr und die Primitive werden folglich als Nichtgebäudeprimitive klassifiziert. Das betrifft z.B. die Eckelemente von Gebäuden mit Satteldächern (vgl. Abbildung 5-16).

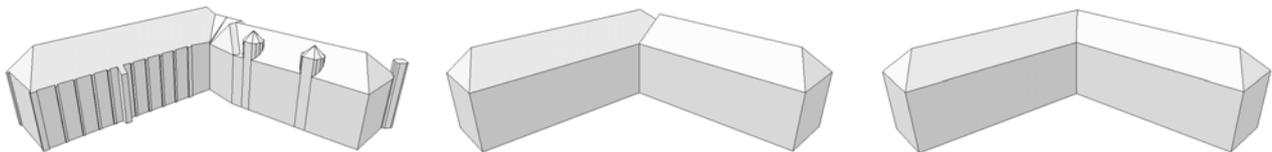


Abbildung 5-16: Original 3D-Gebäudemodell und Generalisierung mit fehlenden und vorhandenen Eckelementen.

In diesen Fällen ist es besser, eine Interpretation der originalen Dachform vorzunehmen, um basierend auf den Ergebnissen eine vereinfachte Geometrie abzuleiten. Dies geschieht wiederum pro Blockprimitiv, da bei deren Erzeugung die Diskontinuitäten in der Dachform schon mit berücksichtigt wurden. Es ist also anzunehmen, dass, obwohl im Bereich des Blockprimitivs verschiedene flache und schräge Dachelemente gemeinsam vorkommen können, genau ein dominanter Dachtyp vorherrschend sein wird.

Um den Dachtyp zu identifizieren, wird das zu untersuchende Blockprimitiv für jede Dachform in eine entsprechende Parameterdarstellung umgewandelt und eine Parameterschätzung durchgeführt. Deren Bewertung erfolgt wieder anhand des diskretisierten Grundrisses. Hierzu wird die Summe der quadratischen Höhenunterschiede zwischen originaler und generalisierter Dachhöhe gebildet, sowie zusätzlich die grobe Orientierung der Dachflächen betrachtet. Der Parametersatz, der ein globales Höhenminimum sowie die meisten richtig orientierten Rasterzellen erzeugt, wird als vorläufiges Ergebnis beibehalten.

Es wird zwischen Sattel-, Walm-, Pult- und Flachdach und den daraus abgeleiteten Sonderformen, asymmetrisches Walmdach, Eck-, T- und Kreuzelement für Satteldächer, unterschieden (siehe Abbildung 5-17). Die Parameterdarstellungen der beiden Dachtypen, Sattel- und Walmdach, benötigen Grundrisse in Form eines Parallelogramms. Der Ansatz ist daher auf Blockprimitive mit dieser Grundform eingeschränkt. In der Praxis zeigt sich jedoch, dass der beschriebene Algorithmus bei der initialen Zerlegung häufig geeignete Primitive erzeugt. Ist dies nicht der Fall, so muss die Dachform mit der vorherigen Vorgehensweise generalisiert werden.

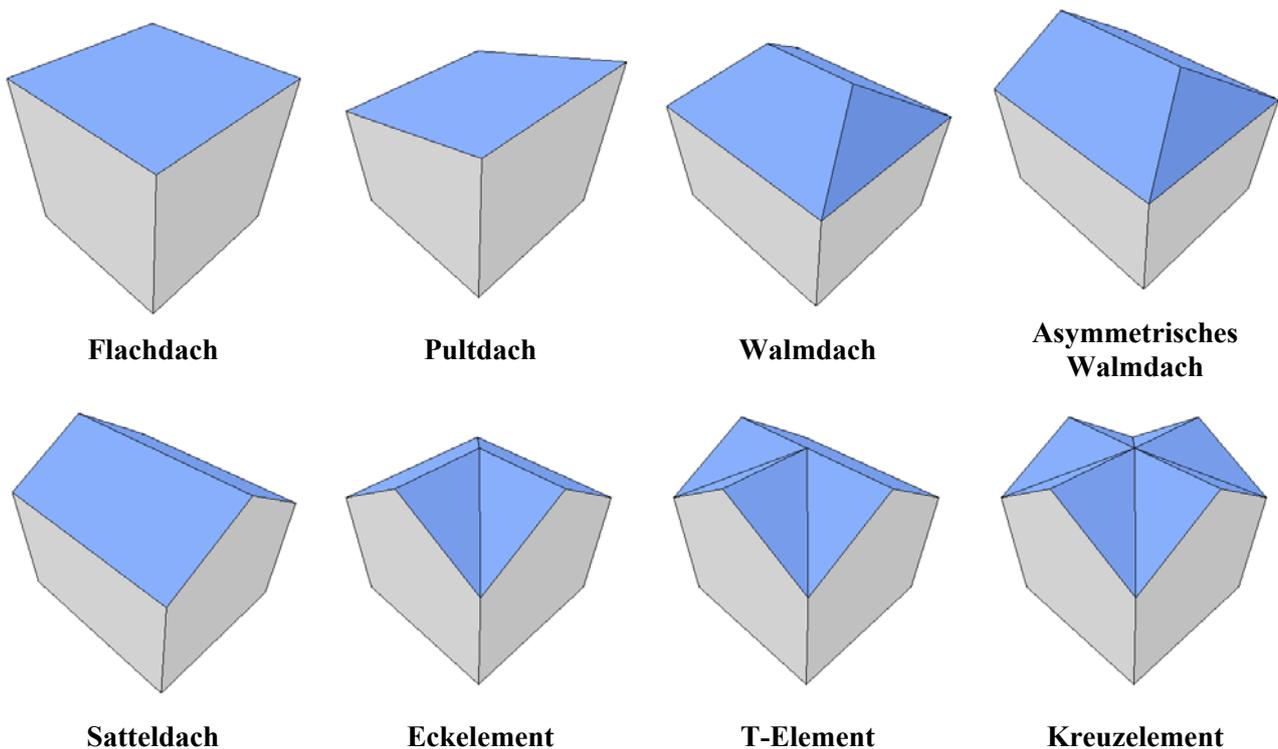


Abbildung 5-17: 3D-Primitive mit den unterstützten Dachformen.

Da die Sonderformen nur für Blockprimitive sinnvoll sind, welche die entsprechenden Nachbarprimitive mit den passenden Dachformen besitzen, erfolgt die Identifikation in aufsteigender Reihenfolge zur Anzahl der Nachbarn. Zusätzlich entscheidet auch die Anordnung der Nachbarprimitive. Tabelle 5-1 gibt die Präferenzen für alle möglichen Situationen wieder. Für jedes Blockprimitiv wird das Ergebnis aus der Parameterschätzung mit den Einträgen in der Tabelle verglichen. Steht an der entsprechenden Stelle ein + oder ++, so gilt der Dachtyp als sicher bis sehr sicher und wird zunächst nicht weiter betrachtet. Steht dort jedoch ein \circ oder gar ein -, so ist das Ergebnis der Parameterschätzung vermutlich oder sogar sehr sicher falsch und es wird nachgeschaut, ob für das Blockprimitiv nicht ein ähnlich guter Parametersatz mit einem anderen Dachtyp existiert. Anschließend wird noch sichergestellt, dass die Grenzflächen von Nachbarprimitive dieselben sind und notfalls korrigiert.

Generell werden bei dieser Vorgehensweise die Satteldachformen bevorzugt, insbesondere wenn die Nachbarprimitive, die selbst eine geringere Anzahl von Nachbarprimitive besitzen, an der Grenzfläche schon ein Satteldach aufweisen. Auf diese Weise bedingen äußere Primitive mit wenigen Nachbarn die angrenzenden, inneren Primitive und es entsteht eine einheitliche Dachstruktur. Die Generalisierung zu Flach- oder Pultdächern ist prinzipiell nie falsch, weswegen all diese Fälle mit einem + recht hoch bewertet sind. Über die Pa-

Parameterschätzung sollten Flachdächer sich deutlich von Sattel- oder Walmdächer unterscheiden lassen. Dies wird darüber hinaus verstärkt, indem der Höhenunterschied für die Parameter Trauf- und Giebelhöhe mindestens der Generalisierungsdistanz entsprechen muss.

# Nachbar-primitive	Anordnung								
0	-	+	+	+	-	+	-	-	-
1	-	+	+	○	+	+	-	-	-
2	gegenüberliegend	+	+	○	-	++	-	-	-
2	über Eck	+	+	-	-	-	++	-	-
3	-	+	+	-	-	-	-	++	-
4	-	+	+	-	-	-	-	-	++
- = weniger geeignet		○ = möglich		+ = gut geeignet		++ = bevorzugt			

Tabelle 5-1 Dachformen in Abhängigkeit von Anzahl und Anordnung der Nachbarprimitive.

Nachdem die Dachtypen sicher identifiziert sind, werden die Parameter aller Blockprimitive nochmals gemeinsam geschätzt, so dass Ungenauigkeiten in der Trauf- und Giebelhöhe eliminiert werden. Beispielhafte Ergebnisse sind in Abbildung 5-14 bis Abbildung 5-16 dargestellt.

Bei den speziellen Dachtypen, Shed-, Parallel- und Grabendach, wiederholen sich gleichförmige Dachelemente. Diese Formen findet man beispielsweise bei Fabrikhallen, Reihenhäusern und in dicht bebauten Innenstädten. Hierfür benötigt es der Definition zusätzlicher Dachtypen. Wurde ein solcher Typ jedoch identifiziert und dessen Parameter bestimmt, so kann die Dachform mittels Typifizieren vereinfacht werden. Zu diesem Zweck wird eine neue Dachform vom selben Typ, aber mit einer geringeren Anzahl von Dachelementen generiert. Etwaige Ränder können dabei mit entfernt werden. Abbildung 5-18 zeigt das Ergebnis der Generalisierung eines Paralleldachs, bei dem die Generalisierungsoperation Typifizieren durch das Ausdünnen von sieben auf fünf Dachelemente realisiert wurde. Dabei wird der Dachtyp durch die drei Parameter Anzahl Traufen, Länge der Traufen und Randbreite beschrieben. Da diese speziellen Dachtypen nur äußerst selten vorkommen, wurden sie nur für Blockprimitive ohne Nachbarn betrachtet, weshalb sie in der Tabelle 5-1 auch nicht berücksichtigt wurden

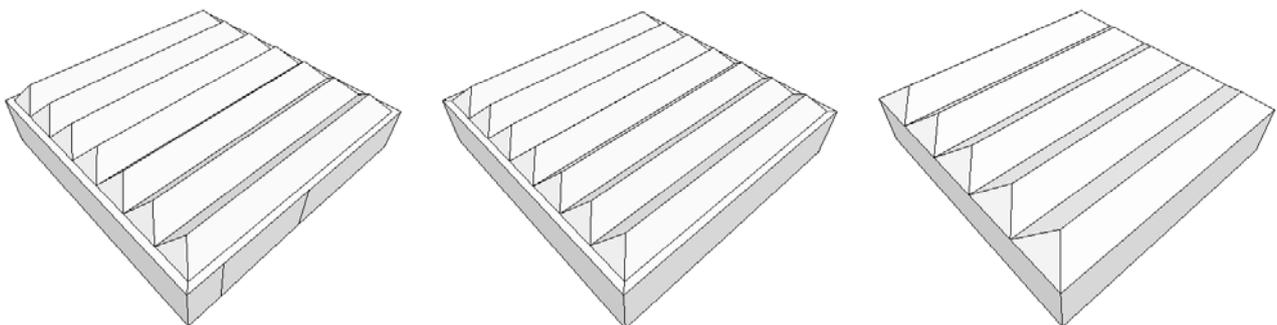


Abbildung 5-18: Originales 3D-Gebäudemodell (links), vereinfachter Nachbau mit denselben Parametern (Mitte) und Ergebnis nach der Ausdünnung der Dachelemente (rechts).

6 Fassadengestaltung

In den Kapiteln 4 und 5 wurde die Transformation von 3D-Gebäudemodellen hin zu einer vereinfachten Form beschrieben. Dies ist für kartenverwandte 3D-Darstellungen von großflächigen Stadtansichten zwar ausreichend; bei einer bodennahen Darstellung wirken die flachen Fassaden jedoch unrealistisch. Es ist daher eine Ausgestaltung der Fassaden notwendig, die hier sowohl bildlich durch die Erzeugung von Fassadentexturen als auch geometrisch aufgezeigt werden soll. Da eine manuelle Bearbeitung für eine gesamte Stadt unrealistisch ist, wird bei den aufgezeigten Verfahren besonders Wert auf die automatische Verarbeitung gelegt. Als Datengrundlage dienen hier terrestrische Bild- und Laserdaten, für welche die äußere Orientierung des Sensors zum Aufnahmezeitpunkt bekannt sein muss. Bevor jedoch auf die automatischen Verfahren eingegangen wird, soll zum besseren Verständnis zunächst kurz der Arbeitsablauf aufgezeigt werden, wie er für die häufig noch manuell durchgeführte Texturierung von Gebäudefassaden notwendig ist.

6.1 Manuelle Texturierung von Gebäudefassaden

Die manuelle Texturierung der Fassaden von 3D-Gebäudemodellen ist äußerst arbeitsintensiv (vgl. Abbildung 6-1). Nach der Aufnahme eines digitalen Photos von der zu erfassenden Fassade muss zunächst eine Korrektur des Bildes durchgeführt werden. Andernfalls erscheinen gerade Gebäudekanten durch die radialen Linsenverzeichnungen der Kamera gebogen und können nur ungenau extrahiert werden. Wurden die Parameter der inneren Orientierung der Kamera durch Kalibrierung bestimmt, so ist eine automatische Korrektur des Bildes mittels entsprechender Softwareprogramme möglich. Anschließend erfolgt die Identifikation viereckiger Bereiche im Bild, die nach der Berichtigung der perspektivischen Verzerrung rechteckige Texturbilder ergeben. War die Fassade zum Aufnahmezeitpunkt durch Fremdobjekte teilweise verdeckt, so sind die betroffenen Texturbereiche noch mittels Bildbearbeitungsprogramme geeignet zu retuschieren.



Abbildung 6-1: Die (manuelle) Texturextraktion umfasst (1) digitale Bildaufnahme und Korrektur der Linsenverzeichnung, (2) Selektion eines viereckigen Bildbereichs, (3) Berichtigung der perspektivischen Verzerrung und (4) Retuschieren von verdeckten Bildbereichen.

Die fertigen Texturbilder können anschließend auf das Gebäudemodell aufgebracht werden (vgl. Abbildung 6-2). Hierzu wird einer polygonalen Fassadenfläche eine Textur zugeordnet. Abhängig vom verwendeten Modellierungspaket ist dabei eine Drehung, Scherung, Skalierung und Verschiebung der Textur möglich. Dadurch kann das Texturbild sehr genau an die Nachbarflächen angepasst werden. Das Ergebnis ist eine Abbildungsvorschrift von den 3D-Koordinaten des Gebäudemodells auf die 2D-Texturkoordinaten. Um Bilddaten einzusparen, lassen sich die Texturen auch mehrfach auf die Gebäudefläche aufbringen, indem z.B. wie in Direct3D und OpenGL Koordinaten außerhalb des Bereichs zwischen Null und Eins verwendet

werden. Das Ergebnis sind texturierte 3D-Gebäudemodelle, wie sie standardmäßig für 3D-Visualisierungssoftware benötigt werden.



Abbildung 6-2: Die manuelle Platzierung des Texturbildes umfasst (1) Selektion des Fassadepolygons, (2) Selektion des Texturbildes, (3) Einstellung der Kachelungsparameter und (4) Feinabstimmung mit Texturen der Nachbarflächen.

Ausgiebige Erfahrungen mit der manuellen Texturierung von 3D-Stadtmodellen konnten im Projekt „Gismo – Echtzeitvisualisierung von virtuellen 3D-Landschaften“ gesammelt werden [Kada et al. 2003]. Ziel des Projekts war die Erstellung eines 3D-Stadtmodells von Stuttgart sowie die Realisierung einer Visualisierungssoftware zu deren interaktiven Exploration. Um ein ansprechendes Erscheinungsbild des Stadtgebiets sowohl im Überflug als auch aus Bodennähe zu garantieren, wurden die Modelle von über 1.000 Gebäuden aus dem Stadtzentrum mittels terrestrischer Bilder erfasst. Der zeitliche Aufwand der manuellen Texturierung betrug insgesamt geschätzte 40 Mannmonate, in denen tausende Photos aufgenommen und zu Fassadentexturen verarbeitet wurden. Um den Arbeitsablauf der manuellen Texturierung weitestgehend zu vereinfachen, wurden für fast alle Arbeitsschritte spezielle Softwareapplikationen entwickelt. Lediglich für das Retuschieren von Fremdobjekten wurde auf Standardsoftware zur Bildbearbeitung zurückgegriffen. Mit etwas Erfahrung im Umgang mit den Werkzeugen war es möglich, ein Gebäudemodell, je nach dessen Komplexität, in einem halben bis zwei Arbeitstagen zu texturieren. Ein häufig wiederkehrendes Problem war, dass Fassadenpolygone aufgrund dichter Bebauung nicht vollständig in einem Photo erfasst werden konnten. Die Texturen mussten dann zeitaufwendig zusammengestückelt werden, wodurch die Qualität, vor allem an den Nahtstellen, gemindert wurde. Die Ergebnisse der manuellen Texturierung sind in Abbildung 6-3 am Beispiel eines Einzelgebäudes und einer Überflugsansicht des Stadtmodells von Stuttgart zu sehen.

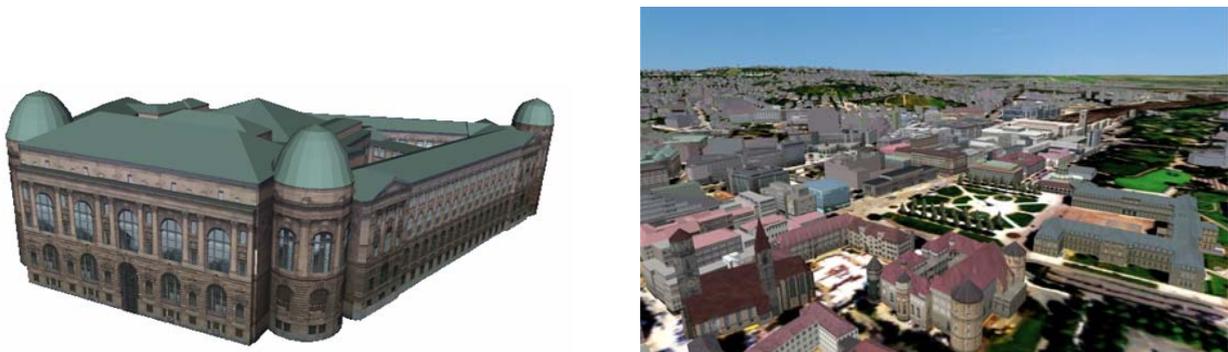


Abbildung 6-3: Ergebnis der manuellen Texturierung am Beispiel eines Einzelgebäudes (links) und in einer Echtzeitvisualisierung des gesamten Stadtmodells von Stuttgart (rechts).

6.2 Automatische Texturierung von Gebäudefassaden

Aufgrund des hohen manuellen Arbeitsaufwands ist eine Automation bei der Texturierung der Gebäudefassaden wünschenswert. Hiefür muss die Korrespondenz zwischen den Weltkoordinaten des georeferenzierten 3D-Gebäudemodells und den Bildkoordinaten des erfassten Photos hergestellt werden, wofür wiederum die exakte äußere Orientierung des Bildes zum Aufnahmezeitpunkt benötigt wird. Klinec und Fritsch [2003] bestimmen die Rotation und Translation der äußeren Orientierung über die Suche nach Korrespondenzen zwischen linienförmigen Objekt- und Bildbestandteilen und verwenden diese in einem photogrammetrischen Rückwärtsschnitt. Anschließend kann aus der inneren und äußeren Orientierung der Kamera eine Transformation T_{WB} berechnet werden, welche die polygonalen Gebäudeflächen ins Bild projizieren, d.h. Weltkoordinaten in Bildkoordinaten umrechnet (siehe Abbildung 6-4).

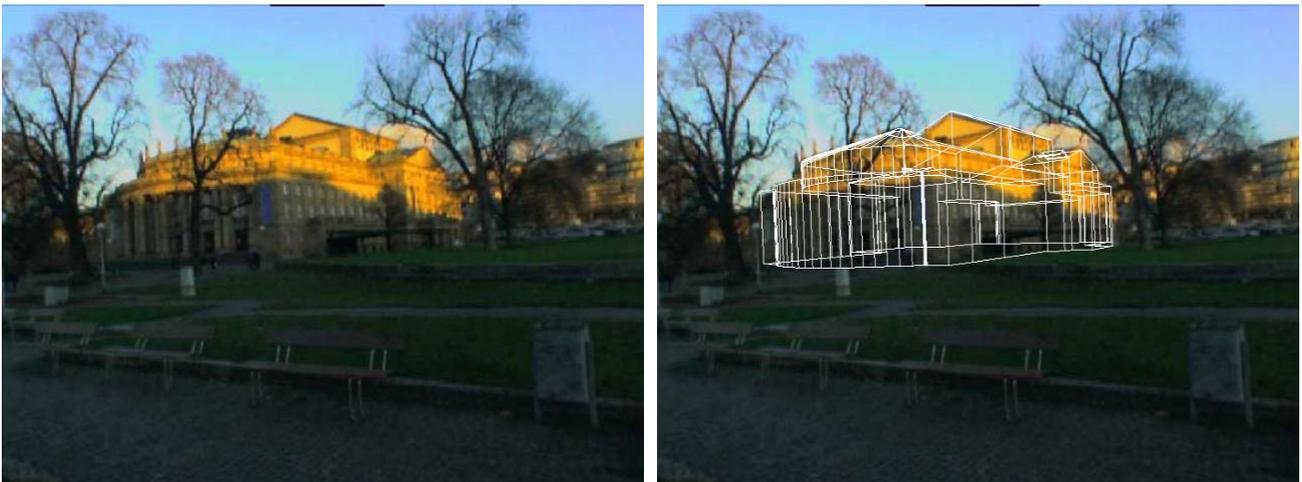


Abbildung 6-4: Photo mit bekannter inneren und äußeren Orientierung (links) und überlagertes georeferenziertes 3D-Gebäudemodell (rechts).

In einem naiven Ansatz zur Texturierung des Gebäudemodells würde man das orientierte Photo direkt als Textur verwenden und die entsprechenden Texturkoordinaten für das Bild mithilfe der oben genannten Transformation T_{WB} berechnen. Eine solche Vorgehensweise hat jedoch einige bedeutsame Nachteile. Der Speicherverbrauch ist recht hoch, da die relevanten Gebäudepixel oftmals nur einen Teil des Photos ausmachen. Es werden auch keine Überdeckungen durch das Gebäude selbst oder durch andere Objekte berücksichtigt, so dass Fassadenflächen mit falschen Farbinformationen texturiert würden. Dies kann nur durch die Kombination von Bildern korrigiert werden, die von verschiedenen Standpunkten aus aufgenommen wurden. Ist eine Fläche in keinem Einzelbild vollständig sichtbar, so muss das Texturbild pixelgenau aus mehreren Bildern zusammengesetzt werden. Ferner können den komplexen geometrischen Bildtransformationen, die durch perspektivische Effekte und Linsenverzeichnungen auftreten, in so einem einfachen Ansatz nicht entgegen gewirkt werden.

Im Folgenden sollen nun die einzelnen Schritte der automatischen Texturextraktion erläutert werden, bevor in Abschnitt 6.2.5 eine Realisierung mittels dedizierter Grafikhardware vorgestellt wird.

6.2.1 Korrektur der Linsenverzeichnung

Wird für die Fassadenaufnahmen eine kalibrierte Kamera verwendet, so können die durch die Linsenverzeichnung hervorgerufenen Störungen im Bild korrigiert werden. In der manuellen Texturierung von Gebäuden gehört dieser Schritt eigentlich zur Vorverarbeitung des Bildes. Wie man in Abschnitt 6.2.5 jedoch sehen wird, kann die Korrektur auch völlig transparent beim Zugriff auf die Farbwerte des Photos realisiert werden. Das hat den Vorteil, dass während der Texturextraktion die Farbwerte aus dem Originalbild ausgelesen werden. Alternativ wird zuerst ein idealisiertes Bild erzeugt, bei deren Berechnung eine bilineare Filterung der Farbwerte erfolgt. Der sich anschließende Extraktionsprozess führt möglicherweise weitere Filteroperationen durch. Jede Filterung der Bilddaten verschlechtert jedoch die resultierenden Fassadentexturen. Um Texturen

von möglichst hoher Qualität zu erhalten, ist es daher ratsam, die Anzahl der Filteroperationen zu minimieren.

Die Linsenverzeichnung wird durch den in [Brown 1971] eingeführten Parametersatz dargestellt, der den Übergang von den Pixelkoordinaten des verzerrten (\bar{x}, \bar{y}) zum idealisierten (x, y) Bild beschreibt.

$$\begin{aligned}x &= \bar{x} + \Delta x \\y &= \bar{y} + \Delta y\end{aligned}$$

Das zur Kalibrierung der Kamera verwendete Bündelprogramm Australis [Fraser 1997] definiert die folgende Untermenge:

$$\begin{aligned}\Delta x &= \Delta x_i + \Delta x_r + \Delta x_d + \Delta x_a \\ \Delta y &= \Delta y_i + \Delta y_r + \Delta y_d + \Delta y_a\end{aligned}$$

Änderung der inneren Orientierung:

$$\begin{aligned}\Delta x_i &= \Delta x_0 - \frac{\bar{x}}{c} \Delta c \\ \Delta y_i &= \Delta y_0 - \frac{\bar{y}}{c} \Delta c\end{aligned}$$

Radial-symmetrische Verzeichnung:

$$\begin{aligned}\Delta x_r &= \bar{x}(r^2 K_1 + r^4 K_2 + r^6 K_3) \\ \Delta y_r &= \bar{y}(r^2 K_1 + r^4 K_2 + r^6 K_3)\end{aligned}$$

Radial-asyymetrische und tangentielle Verzeichnung:

$$\begin{aligned}\Delta x_d &= (r^2 + 2\bar{x}^2)P_1 + 2\bar{x}\bar{y}P_2 \\ \Delta y_d &= 2\bar{x}\bar{y}P_1 + (r^2 + 2\bar{y}^2)P_2\end{aligned}$$

Affinität und Scherung:

$$\begin{aligned}\Delta x_a &= \bar{x}B_1 + \bar{y}B_2 \\ \Delta y_a &= 0\end{aligned}$$

Da sich die Formel nicht invertieren lässt, kann aus dem verzerrten Bild nicht direkt das idealisierte Bild berechnet werden. Daher muss eine iterative Methode verwendet werden, bei der im ersten Schritt die

Gleichheit von verzerrten und idealisierten Bildkoordinaten angenommen wird. In jedem Iterationsschritt werden dann die verzerrten Koordinaten in die obige Formel eingesetzt und die Differenz von Ergebnis- und Eingangskoordinaten von den verzerrten Koordinaten abgezogen. Ist diese Differenz unter einem Pixel, so wurden die verzerrten Koordinaten gefunden, welche durch Einsetzen in die Formel in den idealisierten Koordinaten resultieren. Auf diese Weise kann für jede idealisierte Pixelposition der Farbwert im verzerrten Bild ausgelesen werden, um das idealisierte Bild zu erzeugen.

6.2.2 Texturextraktion und Platzierung

Zweidimensionale Texturen werden üblicherweise als rechteckige Bilddaten gespeichert. Da die planaren Fassadenflächen eines 3D-Gebäudemodells beliebige Polygonformen besitzen dürfen, muss für jede Polygonfläche zunächst ein Minimalumschließendes Rechteck abgeleitet werden. Wird die untere Kante des Rechtecks parallel zur Horizontalebene gesetzt, so ist auch in der resultierenden Fassadentextur der Boden unten. Dies gilt auch für schräge Dachflächen. Um für waagerechte Dachflächen zumindest eine einheitlich Drehung zu erhalten, sollte das Rechteck achsparallel zu den beiden Horizontalachsen ausgerichtet werden. Ist das 3D-Gebäudemodell georeferenziert, so werden die vier Ecken des Minimalumschließenden Rechtecks als dreidimensionale Weltkoordinaten ermittelt. Durch die Verknüpfung der vier Ecken der zu erzeugenden Fassadentextur und denen des Minimalumschließenden Rechtecks kann nun eine Transformation T_{TW} von den Pixelkoordinaten im Texturbildraum in die Weltkoordinaten des globalen Raums berechnet werden. Durch Konkatenation mit der oben bereits erwähnten Transformation T_{WB} erhält man nun eine direkte Transformation T_{TB} von den Pixelkoordinaten des Texturbildes zu den Pixelkoordinaten des Aufnahmebildes. Die Fassadentextur für ein Polygon kann nun zeilenweise Pixel für Pixel gezeichnet werden, wobei die entsprechenden Pixelfarben mittels nächsten Nachbarn oder bilinearer Interpolation aus den Farbwerten des Aufnahmebildes bestimmt werden. Die Größe des Texturbildes lässt sich im Vorfeld bereits aus der projizierten Höhe und Breite des Minimalumschließenden Rechtecks in das Aufnahmebild bestimmen. Der Detailreichtum des Aufnahmebildes wird mit dieser Texturgröße vollständig erfasst. Texturbilder mit größeren Ausmaßen sind jedoch aufgrund externer Anforderungen sinnvoll wie beispielsweise die häufig von Visualisierungssoftware geforderten Seitenlängen in Zweierpotenz.

Nachdem die Fassadentextur erzeugt wurde, muss diese auf das korrespondierende Polygon platziert werden. Dazu sind die beiden Werte der zweidimensionalen Texturkoordinaten als normalisierte Abstände des Eckpunkts jeweils zu einer der beiden vertikalen und einer der beiden horizontalen Kanten des Minimalumschließenden Rechtecks zu berechnen. Die Auswahl der jeweiligen Kanten geschieht in Abhängigkeit vom verwendeten Texturkoordinatensystem. Beispielsweise werden für eine anschließende Visualisierung mit OpenGL die Abstände zur linken und zur unteren Kante benötigt (siehe [Shreiner et al. 2005]). Durch die Normalisierung erhalten die Koordinaten ausschließlich Werte im Bereich zwischen Null und Eins. Falls das Polygon nicht planar sein sollte, so muss der Abstand vom Eckpunkt zu der Ebene genommen werden, die sich aus der Richtung der Kante und des Normalenvektors des Minimalumschließenden Rechtecks ableiten lässt.

Das Ergebnis der automatischen Texturierung ist beispielhaft in Abbildung 6-5 dargestellt. Für die vom Aufnahmestandpunkt sichtbaren Polygonteile des 3D-Gebäudemodells wurden qualitativ hochwertige Fassadentexturen erzeugt. Weil die im Bild nicht sichtbaren Gebäudeteile keine spezielle Behandlung erfuhren, wurde das gesamte Modell texturiert. Dadurch erhielten die vom Aufnahmestandpunkt abgewandten und vom Gebäude selbst verdeckten Gebäudeflächen Texturen mit falschen Farbwerten. Das Verfahren soll daher im nächsten Abschnitt um eine Behandlung von Selbstüberdeckungen erweitert werden.



Abbildung 6-5: Ausgangsbild (links), texturiertes Gebäudemodell von Aufnahmeort betrachtet (Mitte) und aus einer anderen Perspektive (rechts).

6.2.3 Erkennung von Selbstüberdeckungen

Da die Fassade eines Gebäudes mit einem einzigen Photo nicht vollständig erfasst werden kann, sind für eine vollständige Texturierung eines Gebäudemodells mehrere Bilder, die von verschiedenen Standpunkten aufgenommen wurden, notwendig. Voraussetzung für die Texturierung aus mehreren Bildern ist jedoch die Zuordnung von Bildpixeln zu Fassadenpolygonen unter Berücksichtigung von Selbstüberdeckungen. Im vorliegenden Ansatz werden die Überdeckungen während der Texturextraktion zunächst aus einem Einzelbild pixelgenau mithilfe eines zuvor berechneten Distanzbildes bestimmt. Verdeckte Pixel werden dabei in der Fassadentextur mit einem speziellen Farbwert markiert, so dass sich später Texturen aus unterschiedlichen Bildern miteinander zu einer endgültigen Fassadentextur ohne Überdeckungen zusammenfügen lassen.

Hierzu wird zunächst mithilfe des Tiefenpufferalgorithmus (siehe z.B. [Watt 2000]) ein Distanzbild mit derselben Größe wie das Eingabephoto generiert. Darin wird für jedes Pixel die Distanz des am nächsten zum Aufnahmeort gelegenen Gebäudepolygons gespeichert. Anfänglich sind alle Werte auf Unendlich gesetzt. Mittels der Transformation T_{WB} werden die Polygone nacheinander ins Bild projiziert. Dabei ergeben sich für die Eckpunkte eines Polygons zusätzlich zu den Bildkoordinaten auch normalisierte Distanzwerte, die sich über die gesamte Polygonfläche interpolieren lassen. Der neu berechnete Distanzwert eines durch ein Polygon überdeckten Pixels wird mit dem gespeicherten Wert verglichen. Nur wenn die neu berechnete Distanz kleiner ist als der schon gespeicherte Wert, wird das Distanzbild an der Pixelposition aktualisiert. Auf diese Weise verringern sich die Werte im Bereich des Gebäudes schrittweise bis die endgültigen, kürzesten Distanzwerte erreicht werden. Für den Rest des Bildes verbleiben die Werte dagegen auf Unendlich.

Aus der Korrelation T_{TW} zwischen den Pixelkoordinaten im Texturraum und den 3D-Weltkoordinaten des Minimalumschließenden Rechtecks kann während des Extraktionsprozesses für jeden Pixel eine Distanz zum Betrachter berechnet werden. Nur wenn diese Distanz kleiner oder gleich dem Wert im Distanzbild ist, wird der berechnete Farbwert in der Fassadentextur auch gesetzt. Im anderen Fall ist das zu extrahierende Polygon an dieser Pixelposition im Ausgangsbild durch ein anderes Polygon verdeckt und der Texturpixel wird durch einen speziellen Farbwert gekennzeichnet. Der zum Vergleich verwendete Distanzwert wird aus dem Distanzbild auf dieselbe Art und Weise wie der zu setzende Farbwert generiert und durchläuft gegebenenfalls eine bilineare Interpolation. Das Ergebnis der Texturierung mit aktivierter Erkennung von Selbstüberdeckungen ist in Abbildung 6-6 dargestellt. Die im Ausgangsbild nicht sichtbaren Fassadenbereiche sind in der Textur schwarz markiert. Um systematische Bildartefakte, die durch kleine Ungenauigkeiten verursacht werden, zu mindern, wird dem berechneten Wert aus dem Distanzbild vor dem Vergleich ein geringer Epsilon-Wert hinzuaddiert, welcher sich aus dem möglichen Wertebereich für sämtliche Polygone ergibt. Die Artefakte, die an den Polygongrenzen im mittleren Modell in Abbildung 6-6 rot hervorgehoben sind, wurden durch das Setzen des genannten Epsilon-Werts bei der Erzeugung der Texturen für das rechte Bildschirmfoto vollständig eliminiert.



Abbildung 6-6: *Automatisch texturiertes Gebäudemodell mit aktivierter Erkennung von Selbstüberdeckungen vom Aufnahmestandpunkt (links), mit rot hervorgehobenen Artefakten an den Polygongrenzen (Mitte) und aus einer anderen Betrachterposition (rechts).*

6.2.4 Bildfusion

Bis jetzt wurden die Fassadentexturen nur aus einem einzigen Photo extrahiert. Um jedoch mehr als eine Gebäudeseite zu texturieren, müssen Bilder von mehreren Standpunkten aufgenommen und verarbeitet werden. In einem einfachen Ansatz werden Texturen aus allen Bildern erzeugt und anschließend miteinander für ein Fassadepolygon verglichen. Die Textur mit dem geringsten Anteil ungültiger, d.h. überdeckter Fassadenpixel, wird zur Texturierung des Polygons verwendet. Abbildung 6-7 zeigt als Ergebnis zwei Ansichten des texturierten Gebäudemodells, welches aus der Frontalaufnahme aus Abbildung 6-5 und den drei dargestellten Seitenansichten entstanden ist. Gibt es mehr als eine Textur mit ausschließlich gültigen Pixelwerten, so kann man als zusätzliche Auswahlkriterien die Größe der Textur und den damit enthaltenen Detailreichtum oder den Betrachtungswinkel von der Aufnahme position festlegen. Texturen aus frontal photographierten Bildern wirken oftmals natürlicher als aus schrägen Aufnahmen.

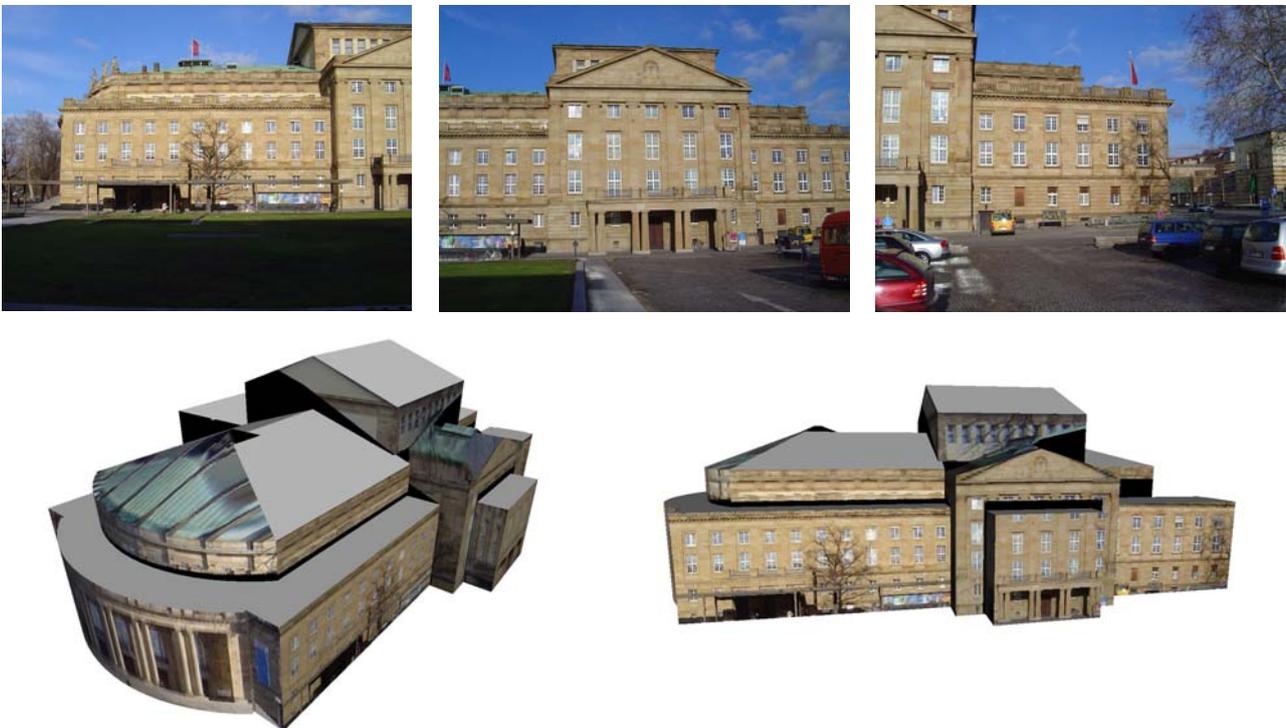


Abbildung 6-7: *Photos von drei Seitenansichten (oben) und daraus texturiertes Gebäudemodell (unten).*

Für sehr große Fassadenpolygone kann es oftmals schwer sein, die gesamte Fläche mit einem einzigen Photo zu erfassen. Ein anderes Problem kann die geringe Auflösung des zu extrahierenden Bildbereichs sein. In solchen Situationen muss die Fassadentextur aus mehreren Bildern Pixel für Pixel zusammengebaut werden.

Die Fassadentextur in Abbildung 6-8 wurde beispielsweise aus zwei Aufnahmen kombiniert, wobei der aus kürzerer Distanz photographierte Frontalaufnahme der Vorzug gegeben wurde. Die fehlenden Texturpixel wurden anschließend mit den Farbwerten aus dem zweiten Bild aufgefüllt. Ein Problem bei einer solchen Vorgehensweise ist die Genauigkeit der äußeren Kameraorientierung. Schon geringe Ungenauigkeiten sind an der Nahtstelle sichtbar.



Abbildung 6-8: Zwei Aufnahmen eines Gebäudes und daraus resultierende Fassadentextur.

6.2.5 Realisierung mit einer Grafikkpipeline

Das vorgestellte Verfahren zur automatischen Erzeugung von Fassadentexturen wurde mittels der Software API Direct3D 9 realisiert. Wie auch OpenGL, verkörpert Direct3D eine 3D-Grafikkpipeline, die aus der vektorialen Beschreibung einer dreidimensionalen Szene gerasterte Bilder in Echtzeit erzeugt. Beide Systeme haben sich in den letzten Jahren von der starren hin zu einer programmierbaren Verarbeitung weiterentwickelt (siehe [Segal und Akeley 2006], [Blythe 2006]). In Kombination mit heute verfügbaren 3D-Grafikkarten sind damit Anwendungen möglich, die weit über das übliche Rendern hinausgehen [Owens et al. 2005]. Die Programmierung der Grafikkpipeline erfolgt über Hochsprachen wie beispielsweise HLSL (High-Level Shader Language) [Gray 2003], Cg (C for Graphics) [Fernando und Kilgard 2003] oder der OpenGL Shading Language [Rost 2006] und ist damit vergleichsweise leicht erlern- und anwendbar. Den Extraktionsprozess von Fassadentexturen auf Basis der Grafikkpipeline Direct3D 9 zu realisieren, ist vor allem darin begründet, dass die Software API den größten Teil der für die Aufgabe notwendigen Funktionalität schon zur Verfügung stellt. Die Kernimplementierung der Texturextraktion beschränkt sich folglich auf wenige Befehle zur Konfiguration der Pipeline. Des Weiteren sind für Direct3D 9 vollständige Implementierungen sowohl in Soft- als auch Hardware vorhanden, was zum einen den Entwicklungsprozess vereinfacht und zum anderen die Ausführung in Echtzeit ermöglicht.

Die Initialisierung der Graphics Pipeline beginnt mit der Spezifizierung einer Pixelfläche, in der das Fassadenbild zu rendern ist. Hierzu wird ein so genanntes Renderziel von der gewünschten Größe und Farbformat erzeugt. Ein entsprechender Farbpuffer wird von der Graphics Pipeline automatisch angelegt und ist fortan Ziel von Zeichenoperationen. Um die gesamte Zielfläche mit Farbwerten zu füllen, wird die Sicht- und Projektionstransformation auf die Identität gesetzt und ein achsenparalleles Rechteck mit Punktkoordinaten von $(-1, -1, 0)$ bis $(+1, +1, 0)$ gerendert. Die vier Eckpunkte des Rechtecks entsprechen dabei den normalisierten Bildkoordinaten des in den Zielraum projizierten Minimalumschließenden Rechtecks des zu extrahierenden Polygons. Zusätzlich werden in den vier Eckpunkten auch die Weltkoordinaten des Minimalumschließenden Rechtecks gehalten, so dass für jedes zu zeichnende Pixel die entsprechende Weltkoordinaten interpoliert werden können. Um das Eingabephoto der Graphics Pipeline zugänglich zu machen, wird dieses in ein Texturobjekt geladen. Der Zugriff erfolgt dann über die interpolierten Weltkoordinaten, die fortan als Texturkoordinaten fungieren. Die Referenzimplementierung von Direct3D 9 erlaubt die Erzeugung von Texturen bis zu einer Größe von $4096 * 4096$ Pixel. Dies entspricht 16 Megapixeln und sollte für die meisten Eingabebilder mehr als ausreichend sein. Durch das Setzen der Texturtransformation mit der bereits erwähnten Transformation T_{WB} können die Farbwerte aus dem Texturbild so ausgelesen werden, dass ein perspektivisch korrektes Abbild der Fassade entsteht. Die Korrektur der Linsenverzeichnung erfolgt dabei transparent beim Zugriff auf die Farbwerte des Eingabephotos. 3D-Grafikkarten, deren Pixelverarbeitung das so genannte Shader Modell 3.0 unterstützen, sind in der Lage, Schleifen mit einem dynamischen Abbruchkriterium zu verarbeiten. Die Linsenverzeichnung kann daher, wie oben beschrieben, iterativ auf der Hardware erfolgen,

indem die idealisierten Texturkoordinaten auf die entsprechenden Koordinaten im verzerrten Bild umgerechnet werden. Das resultierende Fassadenbild kann nach dem Rendern aller Pixelwerte aus dem Farbpuffer ausgelesen und gespeichert werden.

Für die Erkennung von Selbstüberdeckungen wird ein Distanzbild von derselben Größe wie das Eingabephoto benötigt. Hierzu werden die Polygone des Gebäudemodells, wie oben beschrieben, mit aktiviertem Tiefenpuffer gerendert. Der Inhalt des Tiefenpuffers kann anschließend in eine Textur kopiert werden, die pro Pixel einen 32-Bit Fließkommawert verwaltet. Als effiziente Alternative kann der Tiefenwert mithilfe eines Programms in der Pixelverarbeitung der Grafikkpipeline auch direkt in den 32-Bit Farbpuffer der Distanztextur gerendert werden. Ein weiteres Pixelprogramm vergleicht während der Texturextraktion dann den berechneten Distanzwert mit dem in der Textur gespeicherten Wert. Ist der gespeicherte Wert kleiner als die berechnete Distanz, so wird das Polygon an dieser Pixelposition durch ein näher am Aufnahmestandpunkt gelegenes Polygon überdeckt. Der betroffene Pixel wird in der Fassadentextur im Hinblick auf die sich anschließende Bildfusion entsprechend markiert. Dies kann beispielsweise im Alphakanal des Farbpuffers geschehen. Der Farbpuffer wird anfangs mit dem Alphawert Null initialisiert, der für ein verdecktes Pixel steht. Eine Aktualisierung von Pixelfarben findet nur statt, wenn der neue Alphawert gleich oder größer dem gespeicherten Wert ist. Sichtbare Pixel erhalten den Alphawert Eins und werden folglich immer aktualisiert. Ein verdeckter Pixel erhält stattdessen den Alphawert Null. Da eine falsche Pixelfarbe einer fehlenden Farbe unter Umständen vorzuziehen ist, wird die Pixelfarbe nur verändert, wenn der gespeicherte Alphawert Null ist. Der Zugriff auf die Zwischenergebnisse wird nach der Prozessierung jedes Eingabebildes über eine Textur dem Extraktionsprozess wieder zur Verfügung gestellt. Das Resultat ist eine Fassadentextur, in der die verdeckten Pixel mittels des Alphawerts korrekt identifiziert werden können. Das Ablaufdiagramm der Texturzeugung ist in Abbildung 6-9 dargestellt.

6.2.6 Bewertung

Die Implementierung der automatischen Texturextraktion mithilfe der Grafikkpipeline Direct3D 9 hat sich als vergleichsweise unkompliziert herausgestellt. Alle für die Aufgabe benötigten Vektor- und Rasteroperationen sind in heutigen Grafikkpipelines vorhanden und lassen sich über die gut dokumentierte Schnittstelle nutzen. Oft wiederkehrende Funktionalitäten, wie beispielsweise das Laden, Speichern und Rendern in Texturen sind, dank der D3DX Utility-Bibliothek, mit nur einem Funktionsaufruf möglich. Die Verarbeitungsgeschwindigkeit der Software wurde auf einem Standard-PC gemessen, dessen Grafikkchip das Shader Modell 3.0 unterstützt. Sowohl die maximal 16 verwendeten Eingabebilder, als auch die Gebäudemodelle mit 71 und 149 Polygonen fanden vollständig im Speicher der Grafikkarte Platz. Die Ergebnisse sind in Abbildung 6-7 bis Abbildung 6-8 zu sehen. Da sich die Leistungsfähigkeit der Grafikkhardware rasant weiterentwickelt, soll auf die genauen Messwerte an dieser Stelle verzichtet werden. Wie erwartet, konnte ungefähr eine lineare Abhängigkeit sowohl von der Anzahl der verwendeten Eingabebilder als auch der zu extrahierenden Polygone beobachtet werden. Für die Texturextraktion des gesamten Gebäudes aus einem Bild wurde für die beiden Modelle eine Verarbeitungszeit zwischen 40 und 80 ms gemessen. Selbst bei 16 Eingabebildern brauchte das System nur zwischen 400 und 800 ms. Damit sind interaktive Anwendungen möglich, bei dem der Operateur Photos zum Prozess hinzufügt oder entfernt und die Auswirkungen sofort anhand des Ergebnisses überprüfen kann.

Die automatische Texturierung von 3D-Gebäudemodellen hat generell mit zwei Problemen zu kämpfen. Da wäre zum einen die Genauigkeit der ermittelten äußeren Orientierung der Photos und zum anderen etwaige Inkonsistenzen zwischen Modell und Realität. Inkonsistenzen resultieren zumeist aus der Genauigkeit, mit dem das verwendete Rekonstruktionsverfahren die Lage und Form des Gebäudes exakt nachzubilden vermag. Wird die Berechnung der äußeren Orientierung zudem über eine Zuordnung von Bild und 3D-Modell gelöst, so wirken sich diese Inkonsistenzen wiederum negativ auf die Genauigkeit der äußeren Orientierung aus. Das Ergebnis sind Fassadentexturen mit Pixel an den Rändern, die nicht zum Gebäude gehören, sondern stattdessen die nähere Umgebung des Gebäudes widerspiegeln. Vor allem die blauen Farbtöne des Himmels unterscheiden sich oftmals von der gängigen Farbgestaltung von Gebäuden und werden daher vom Betrachter schnell als störend wahrgenommen. In einer semi-automatischen Texturextraktion kann der Operateur die Objektpunkte an den terrestrischen Bildern manuell nachmessen, um die Übereinstimmung mit der Realität zu verbessern. Die hardwarebasierte Texturextraktion erzeugt sofort ein texturiertes Gebäudemodell, an dem der Operateur die geometrischen Veränderungen überprüfen kann.

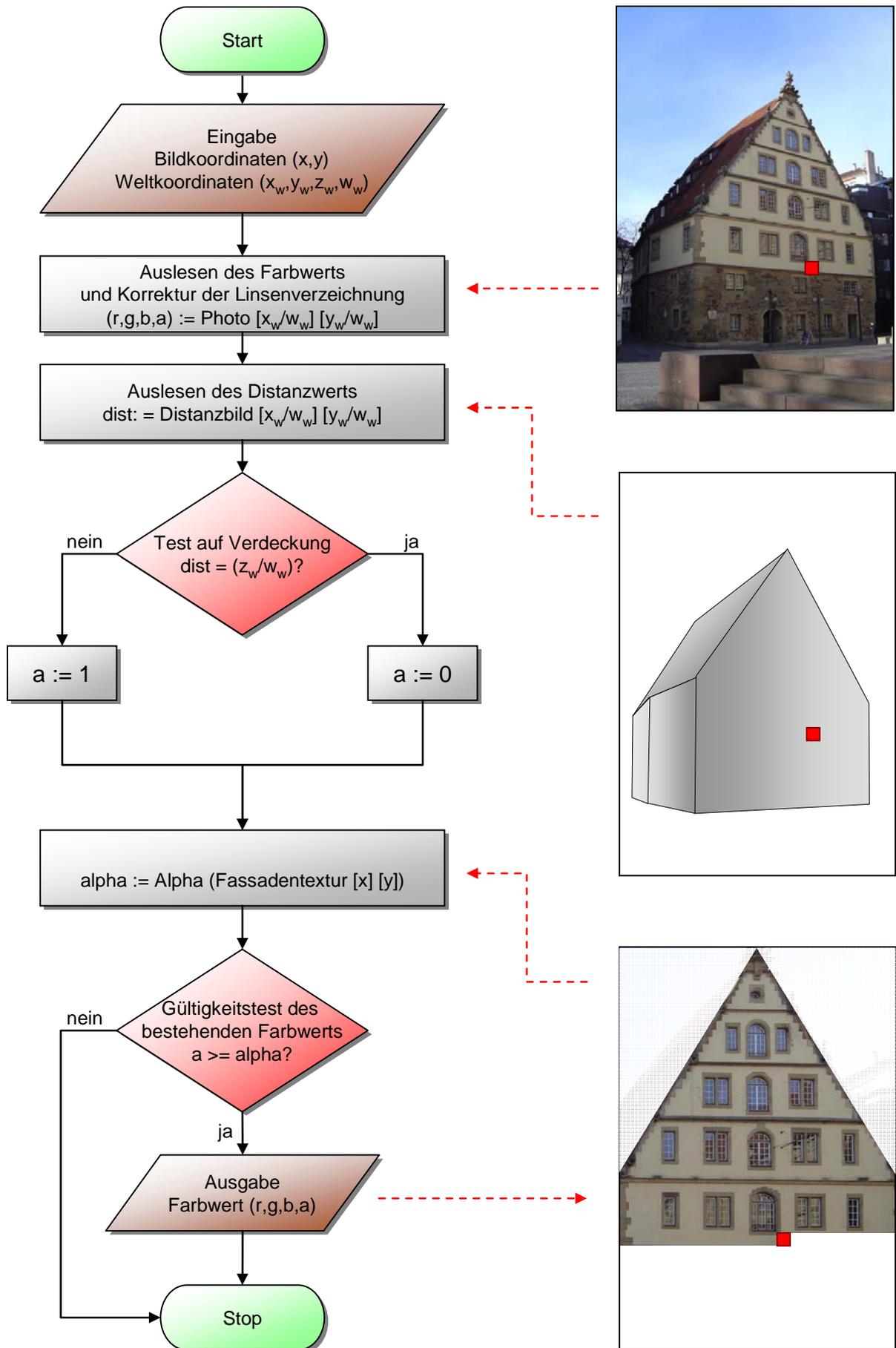


Abbildung 6-9 Ablaufdiagramm der automatischen Texturerzeugung.

Das Verfahren behandelt bisher ausschließlich Selbstüberdeckungen, bzw. es können nur Fremdoobjekte in den Verarbeitungsprozess integriert werden, die als 3D-Modelle zur Verfügung stehen. Dies ist aber vor allem bei kleinen oder dynamischen Objekten (z.B. Denkmäler, Verkehrsteilnehmer, etc.) nur selten der Fall. Denkbar ist daher eine Erweiterung des Verfahrens, um Störobjekte bildbasiert zu erkennen und zu entfernen. Dies kann beispielsweise über eine Folge von Einzelbildern realisiert werden, die zu unterschiedlichen Zeitpunkten oder von verschiedenen Standpunkten aus aufgenommen wurden. In jedem Einzelbild sollte die Fassade zumindest teilweise sichtbar sein, so dass sich aus mehreren Bildern eine verdeckungsfreie Ansicht konstruieren lässt. Böhm [2004] berechnet hierzu zunächst eine pixelgenaue Registrierung der Bilder, indem auf der relevanten Fassadenfläche jeweils vier identische Punkte gemessen werden. Daraus lassen sich perspektivische Transformationen ableiten, welche die Bildbereiche der planaren Fassade in eine gemeinsame Projektion überführen. Mittels eines sich anschließenden Clusterverfahrens kann nun pixelweise aus allen Bildern der korrekte Farbwert des statischen Hintergrunds ausgewählt werden, wodurch sich eine Fassadentextur, ergibt die frei von Störobjekten ist (siehe Abbildung 6-10). Ortin und Remondino [2005] beschreiben ihre Erfahrungen mit einem nahezu identischen Ansatz. Die Farbe eines Fassadenpixels wird jedoch nicht durch den wahrscheinlichsten Wert bestimmt, sondern berechnet sich als Medianwert aus allen korrespondierenden Pixeln. Es wird berichtet, dass die resultierenden Fassadenbilder für beide Auswahlkriterien dieselbe Qualität erreichen. Wang et al. [2002] interpretieren die Anordnung von Fenstern, um verdeckte Regionen zu identifizieren und durch Bildinhalten aus anderen Fassadenbereichen zu vervollständigen.



Abbildung 6-10: Vier Fassadenbilder, aufgenommen von einem festen Standpunkt auf der gegenüberliegenden Straßenseite. Verkehr und Fußgänger verdecken den unteren Teil der Fassade (links). Das resultierende Bild enthält nur den statischen Hintergrund (rechts). (aus [Böhm 2004])

6.3 Fassadenrekonstruktion

Die bildliche Ausgestaltung eignet sich hervorragend, um Material und Kleinststrukturen von weitgehend flachen Fassadenbereichen darzustellen. Für Gebäudeteile, die hervorstehen oder ins Gebäude hineinragen, sind Texturen jedoch nicht ausreichend, da diese bei naher Betrachtung von einer vom Aufnahmestandpunkt abweichenden Blickposition unwirklich erscheinen. Sofern beispielsweise der Fensterrahmen nicht in einer Front mit der Fassade abschließt, sind in Abhängigkeit von der Betrachterposition verschiedene Laibungen sichtbar. Eine geometrische Ausgestaltung der heutzutage gängigen 3D-Gebäudemodelle ist daher für eine realistische Präsentation unumgänglich. Hierfür bieten sich als Ausgangsdaten vor allem terrestrische Laserscans an. Aus diesen generieren Böhm und Haala [2005] z.B. flächenbezogene Laserdistanzbilder für die Verwendung als Bump- oder Displacement Maps. Da es sich dabei um 2D-Abbildungen handelt, können diese Techniken jedoch eine echte geometrische Ausgestaltung nicht ersetzen. Daher soll ein Verfahren zur

Fassadenrekonstruktion vorgestellt werden, welches auf der Zellenzerlegungsmethode aus Kapitel 5 basiert. Aus den uninterpretierten Punktwolken der Fassaden werden Fenster und Türen automatisch detektiert und modelliert. Da sowohl Türen als auch Fenster Fassadeneinbuchtungen mit vergleichbaren Ausmaßen sind, werden Türen nicht gesondert behandelt. Bei der Ausgestaltung der Fenster werden die Türen ebenfalls erzeugt.

6.3.1 Vorverarbeitung

Der terrestrische Laserscanner liefert räumliche Informationen in Form eines unregelmäßigen 3D-Punktehaufens. Vor der Rekonstruktion einer Gebäudefassade ist daher eine Zuordnung von Punkten zu Fassadenseiten notwendig. Diese kann manuell unter Zuhilfenahme von Auswahlwerkzeugen in Softwareapplikationen wie Cyclone oder PolyWorks erfolgen. Komfortabler ist jedoch eine automatische Selektion der entsprechenden Punkte, indem zusätzliche geometrische Informationen, wie beispielsweise der 2D-Gebäudegrundriss, genutzt werden. Ein 3D-Punkt wird (unter Nichtberücksichtigung der z-Koordinate) dem in der Grundrissebene am nächsten gelegenen Liniensegment zugeordnet. Aus der Punktmenge eines jeden Segments wird dann separat eine Fassadenseite rekonstruiert. Für detaillierte Grundrisse bietet sich jedoch vorher eine 2D-Grundrissgeneralisierung an. Dies verringert die Anzahl der Liniensegmente. Die Rekonstruktion erfolgt nun für weniger, dafür aber größere Fassadenflächen. Wiederkehrende Fassadenelemente können so einfacher erkannt und angepasst rekonstruiert werden. Ein einfaches Beispiel sind einheitliche Fensterreihen. Die Größe und Lage von Fenstern auf demselben Stockwerk sind oftmals einheitlich. Je mehr Fensterobjekte auf einmal rekonstruiert werden, desto einfacher können solche Fassadeneigenschaften mit berücksichtigt werden. Die durch die Grundrissgeneralisierung verlorene 2D-Geometrie sollte durch die Rekonstruktion wieder erzeugt werden. Dabei können die Liniensegmente des ursprünglichen Grundrisses mit genutzt werden, um die Fassadenelemente besser zu erkennen. Nachfolgende Ausführungen beziehen sich auf die Rekonstruktion einer einzelnen Fassadenseite.

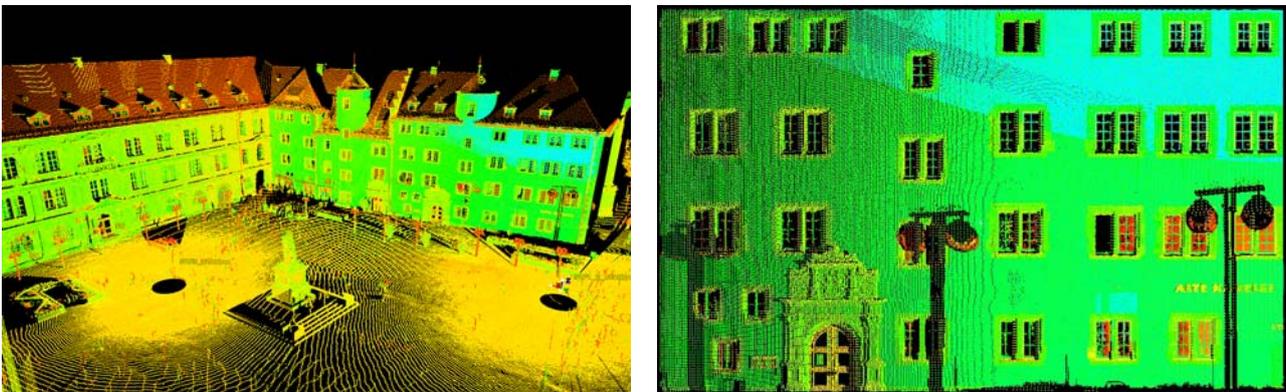


Abbildung 6-11: Durch terrestrisches Laserscanning erfasste 3D-Punktwolke vom Stuttgarter Schillerplatz (links) und selektierte Punkte für die Rekonstruktion einer Fassade der Alten Kanzlei (rechts).

Im ersten Schritt der Rekonstruktion wird aus den 3D-Koordinaten der selektierten Messpunkte die Referenzebene für die Fassadenseite abgeleitet. Damit die Ebene einen großen Bereich der Fassadenseite gut approximiert, sollten möglichst viele Punkte in ihrer Nähe liegen. Aus den weiter entfernten Punkten werden anschließend die Fassadenelemente modelliert. Eine Schätzung der Referenzebene kann über die iterative Eigenwertzerlegung der Matrix der zweiten zentralen Momente erfolgen. Für den Schwerpunkt \bar{x} aus n Messpunkten

$$\bar{x} = \frac{1}{n} \sum_{i=0}^{n-1} x_i$$

wird die Matrix der zweiten zentralen Momente berechnet:

$$M_x = \begin{pmatrix} \sum_{i=0}^{n-1} (x_i - \bar{x})^2 & \sum_{i=0}^{n-1} (x_i - \bar{x})(y_i - \bar{y}) & \sum_{i=0}^{n-1} (x_i - \bar{x})(z_i - \bar{z}) \\ \sum_{i=0}^{n-1} (x_i - \bar{x})(y_i - \bar{y}) & \sum_{i=0}^{n-1} (y_i - \bar{y})^2 & \sum_{i=0}^{n-1} (y_i - \bar{y})(z_i - \bar{z}) \\ \sum_{i=0}^{n-1} (x_i - \bar{x})(z_i - \bar{z}) & \sum_{i=0}^{n-1} (y_i - \bar{y})(z_i - \bar{z}) & \sum_{i=0}^{n-1} (z_i - \bar{z})^2 \end{pmatrix}$$

Auf die Einheitslänge normiert, stellt der zum kleinsten Eigenwert der Matrix M_x gehörende Eigenvektor den Normalenvektor \vec{n} der geschätzten Referenzebene dar. Weil der Schwerpunkt \vec{x} auf der Ebene liegt, erhält man mit $d = -\vec{n} \cdot \vec{x}$ die Hesse'sche Normalform

$$\vec{n} \cdot \vec{x} + d = 0$$

Die geschätzte Ebene minimiert die senkrechten Abstände der Punkte. Weil alle Punkte gleich gewichtet werden, ist die Methode jedoch anfällig gegen Ausreißer. Daher wird die Ebene iterativ berechnet, bis alle Punkte innerhalb einer voreingestellten Schwellwertdistanz liegen. Dabei werden nach jeder Iteration die von der Ebene weit entfernten Punkte von der weiteren Berechnung ausgeschlossen.

Eine weitere, robuste Ebenenschätzung ist beispielsweise auch mithilfe der RANSAC (Random Sample Consensus)-Methode [Fischler und Bolles 1981] möglich. Dabei wird aus drei zufällig gewählten Punkten eine Referenzebene berechnet, bis sich eine gute Lösung ergibt. Eine Bewertung für die gefundene Ebene erfolgt über die Anzahl der Messpunkte, die sich unter der Schwellwertdistanz zur Ebene befinden. Für gleiche Schwellwerte liefern beide Methoden ähnliche Ergebnisebenen.

Vor der weiteren Prozessierung wird die 3D-Punktewolke in das lokale Koordinatensystem der Referenzebene transformiert. Die x, y-Koordinaten der Punkte gibt die Position auf der Referenzebene an. Deren Entfernung zur Ebene ist durch die z-Komponente gegeben.

6.3.2 Segmentierung von Fensterkanten

Die Erkennung von Fensterkanten basiert auf der Annahme, dass sich die an Fenstern gemessenen Laserpunkte hinter der Referenzebene befinden. Punkte an den Fensterkanten liegen also auf oder vor der Referenzebene und besitzen zur Fensterseite hin nur Nachbarpunkte hinter der Referenzebene. Um die Detektion dieser Punkte zu vereinfachen, werden alle Punkte vor der Referenzebene auf die Ebene gezogen und alle Punkte hinter der Referenzebene vorerst entfernt. Die gesuchten Punkte, die in obere, untere, rechte und linke Kantenpunkte unterschieden werden, besitzen innerhalb eines Suchradius daher keine Nachbarpunkte in der entsprechenden Richtung. Auf diese Weise können jedoch nur Kantenpunkte gefunden werden, die an strikt horizontalen oder vertikalen Fensterkanten liegen. Rundbögen, die vor allem bei oberen Fensterkanten auftreten, werden an dieser Stelle zunächst ignoriert. Für jeden Punkt im Datensatz wird ermittelt, in welche der vier Richtungen Nachbarpunkte vorhanden sind, die sich innerhalb des Suchradius befinden. Ein linker Kantenpunkt wird beispielsweise detektiert, wenn innerhalb des Suchradius weitere Punkte links, aber nicht rechts von dem Punkt befinden. Obere, untere und rechte Kantenpunkte werden analog detektiert. Grafisch dargestellt ergeben sich vier Masken für die untersuchten Kantenpunkte deren entsprechende Halbkreise, im Bild schraffiert gezeichnet, frei von weiteren Punkten sein müssen (siehe Abbildung 6-12). Als Suchradius wird für Kantenpunkte ungefähr das Zweieinhalbfache des Abtastabstands bezogen auf die Referenzebene der Fassade verwendet. Bei einem kleineren Suchradius läuft man in Bereichen mit geringer Punktdichte Gefahr, Punkte fälschlicherweise als Kantenpunkte zu detektieren.

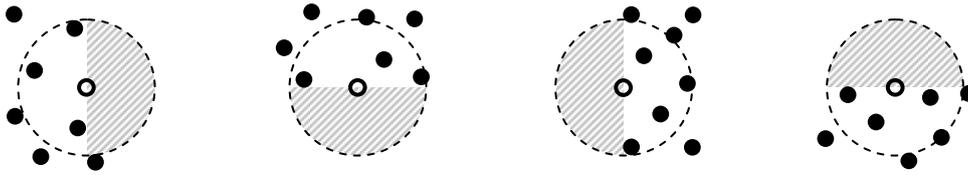


Abbildung 6-12: Masken für linke, obere, rechte und untere Kantenpunkte.

Nahe Fensterecken können die Kantenpunkte nicht mit der dargestellten Methode detektiert werden, da sich im entsprechenden Halbkreis auch immer Punkte einer zweiten Kante befinden. Der Suchbereich darf in diesen Bereichen also nicht so restriktiv sein. Für die Detektion eines Kanten- bzw. Eckpunkts genügt hier ein freier Quadrant in der Suchmaske (siehe Abbildung 6-13). Der Suchradius wird jedoch auf das Fünffache des Abtastabstands erhöht, um Fehlklassifikationen auszuschließen. Auf diese Weise können weitere Punkte in den Fensterecken korrekt als Kantenpunkte identifiziert werden. Die exakten Eckpunkte müssen an dieser Stelle nicht gefunden werden, da diese sich später durch den Schnitt der Fensterkanten automatisch ergeben. Wichtig ist nur eine genügend hohe Anzahl von Kantenpunkten zu detektieren, um daraus die vertikalen und horizontalen Fensterkanten zu mitteln.



Abbildung 6-13: Masken für Kantenpunkte nahe der vier Fensterecken.

Um die Suche nach Kantenpunkten zu beschleunigen werden die Punkte in sortierten Listen verwaltet, die den Raum in überlappende Zeilen und Spalten aufteilen. Die Höhe der Zeilen bzw. Breite der Spalten ist gleich dem dreifachen Suchradius. Bei einer zwei Drittel Überlappung der Suchbänder befinden sich für die Punkte im mittleren Drittel alle weiteren zu deren Evaluation benötigten Punkte in derselben Liste. Durch Sortierung der Zeilen in x-Richtung und der Spalten in y-Richtung muss jeweils nur der Abstand zweier aufeinander folgender Punkte einer Liste verglichen werden. Die Prüfung auf rechten und linken Kantenpunkt bzw. oberen und unteren Kantenpunkt kann dabei gleichzeitig erfolgen. Die Klassifikation der Punkte im Randbereich wird jedoch nicht gespeichert, da sie nur in dem entsprechenden Suchband korrekt detektiert werden können. Auf diese Weise ist ein effizienter Test der Punkte möglich. Die in Abbildung 6-14 dargestellten Kantenpunkte konnten in nur wenigen Sekunden aus den 235.368 Punkten des Beispieldatensatzes segmentiert werden. Dabei wurden die meisten falsch detektierten Kantenpunkte anschließend entfernt, indem der Abstand zum nächsten Kantenpunkt betrachtet wurde. Ist dieser Abstand zu groß, wird der Punkt nicht als Kantenpunkt klassifiziert. Falsch detektierte Punktpaare bleiben dadurch jedoch erhalten. Ein geringes Vorkommen solcher Punkte stört die weitere Verarbeitung aber nicht.

6.3.3 Zellenzerlegung

Aus den x- und y-Koordinaten der Kantenpunkte werden nun Teilungsebenen für die Zellenzerlegung der Fassadenseite abgeleitet. Die vertikalen Teilungsebenen ergeben sich aus den linken und rechten Kantenpunkten durch Gruppieren der x-Koordinaten und der daraus berechneten Durchschnittswerte. Die horizontalen Teilungsebenen werden analog aus den y-Koordinaten der oberen und unteren Kantenpunkte ermittelt. Um die Fensterbereiche aus der Fassade auszustanzten, wird zusätzlich noch eine Teilungsebene parallel zur Referenzebene benötigt. Deren Position ist durch die Fenstertiefe gegeben, die durch das schrittweise Verschieben einer Ebene von der maximalen z-Position hin zur Referenzebene ermittelt werden kann. Für jede Position wird die Standardabweichung zu den gemessenen Punkten berechnet. Dabei werden nur die Punkte hinter der Fassadenebene betrachtet. Da im Allgemeinen mehr Punkte auf der Fensterfläche als auf der Fensterlaibung gemessen werden, ergibt sich die geringste Abweichung im Bereich der gesuchten Fenstertiefe.

Nachdem die Teilungsebenen gefunden wurden, erfolgt nun die Zellenzerlegung der Fassade. Sofern schon ein grobes 3D-Gebäudemodell existiert, kann dieses nach einer Transformation ins lokale Koordinatensystem der Referenzebene verwendet werden. Dabei sollte darauf geachtet werden, dass durch die Transformation die entsprechende Gebäudeseite genau auf der Referenzebene liegt. Alternativ kann aber auch ein aus der Punktwolke abgeleiteter, achsenparalleler Hüllquader als Ersatzgeometrie dienen. Sollen die Fensterflächen mit modelliert werden, so muss die Tiefe des Quaders jedoch auf die eineinhalbfache bis doppelte Fenstertiefe erweitert werden. Die Zellenzerlegung erfolgt nun durch Schnitt des Gebäudemodells bzw. der Ersatzgeometrie entlang der Teilungsebenen, wodurch quaderförmige 3D-Körper entstehen (siehe Abbildung 6-15).

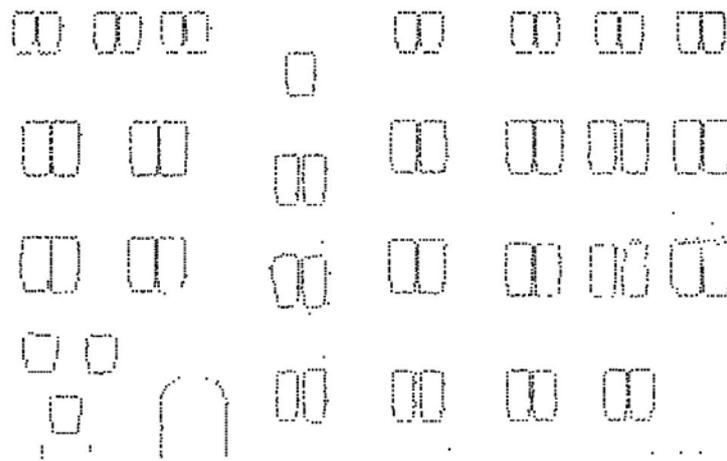


Abbildung 6-14: Detektierte Punkte an den Fensterkanten der Gebäudefassade.

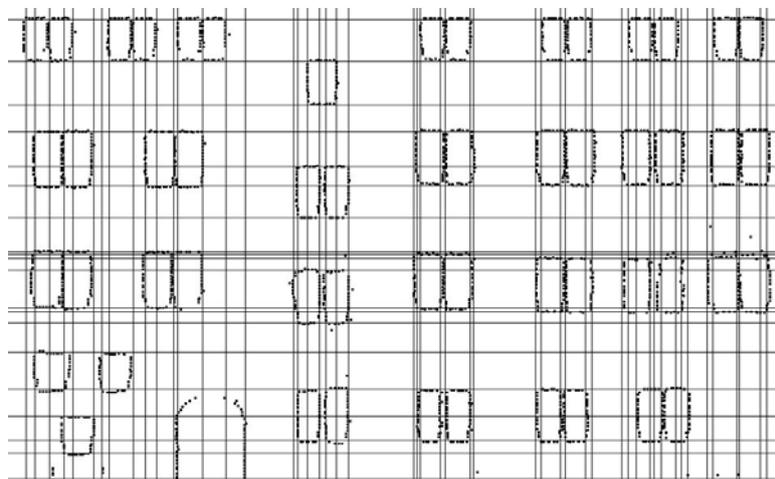


Abbildung 6-15: Zellenzerlegung der Fassadenseite mit überlagerten Kantenpunkten.

6.3.4 Grobklassifizierung

Nach der Erzeugung der Zellenzerlegung werden die 3D-Zellen nun in Fassaden- und Fensterzellen klassifiziert. Fassadenzellen sind Teil der Gebäudewand, d.h. deren Vorderseite ist Bestandteil der Fassadenoberfläche und damit liegen die Zellen hinter der eigentlichen Fassadenfront. Diese Zellen bleiben daher im fertigen Modell erhalten. Fensterzellen hingegen verkörpern den leeren Raum vor den Fenstern und müssen demzufolge aus dem Gebäudemodell entfernt werden.

Die Klassifizierung einer Zelle erfolgt über die prozentuale Abdeckung der Vorderfläche mit gemessenen Fassadenpunkten. Dabei werden sowohl Punkte auf und vor der Referenzebene verwendet, da eine hohe Abdeckung für beide Punktarten einen Festkörper in diesem Raumsegment postulieren. Aufgrund der unregel-

mäßigen Verteilung der per terrestrischen Laserscanner erfassten Punktwolke ist die Berechnung der Abdeckung aus den Rohdaten jedoch nicht so einfach möglich. Daher wird aus den 3D-Punkten mittels einer Flächenabtastung ein 2D-Binärrasterbild erzeugt. Jede Rasterzelle repräsentiert eine gleichgroße Fassadenregion. Der Binärwert gibt an, ob sich ein Messpunkt in diesem Bereich auf oder vor der Referenzebene befindet. Folglich sind in den Fensterbereichen die Rasterzellen weitestgehend leer. Die Rasterzellengröße ist wieder das Zweieinhalbfache des Abtastabstands bezogen auf die Referenzebene. Bei einer kleineren Rasterzellengröße entstehen ansonsten in Fassadenbereichen mit geringer Punktdichte zu viele leere Zellen. Bei einer geringen Fehlerrate können solche falschen Rasterwerte mit dem morphologischen Operator Schließen wieder korrigiert werden. Das Ergebnis der Abtastung ist in Abbildung 6-16 zu sehen.

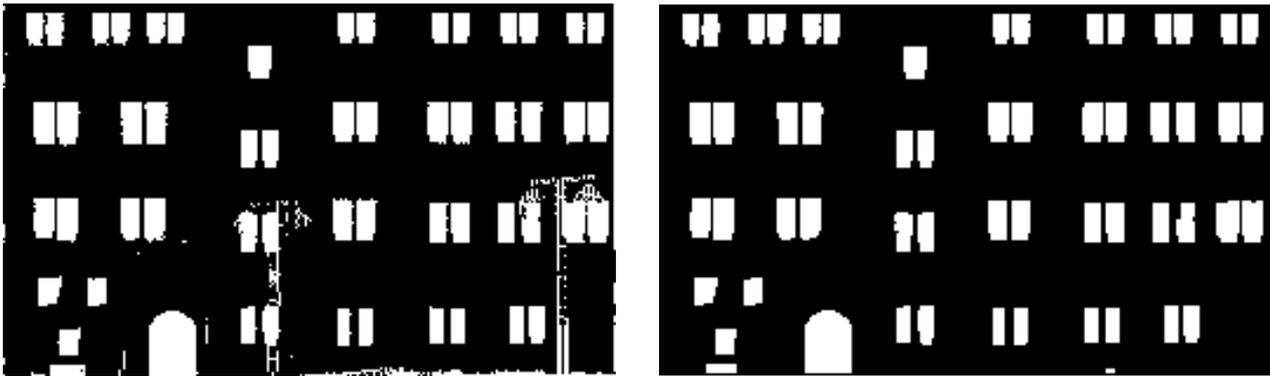


Abbildung 6-16: Binäres Abdeckungsbild vor und nach Anwendung des morphologischen Filters Schließen.

Das Verhältnis gesetzter zu nicht gesetzter Rasterwerte im Fassadenbereich einer Zelle entscheidet über deren Klassifizierung in Fassaden- oder Fensterzelle. Aufgrund der groben Rastereinteilung stimmen die Fensterkanten nur äußerst selten mit den Rastergrenzen überein. Daher erfolgt die Klassifizierung der 3D-Zellen mittels einer quadratischen Gewichtungsfunktion. Diese gewichtet die inneren Rasterzellen höher als die Rasterzellen am Rand. Ungenauigkeiten an den Fensterkanten im Rasterbild fallen dadurch nicht so stark ins Gewicht. Eine Zelle wird als Fassadenzelle klassifiziert, wenn das Rasterverhältnis von Fassaden- zu Nichtfassadenpixel größer oder gleich 70% ist. Bei einem Verhältnis von unter 10% wird eine Fensterzelle klassifiziert. Diese strikten Schwellwerte garantieren eine äußerst zuverlässige Klassifizierung von Zellen, deren Rasterverhältnisse in diese Wertebereiche fallen. Für die restlichen Zellen ist die Klassifizierung unsicher und es werden weitergehende Untersuchungen der betroffenen Zellen notwendig. Das Ergebnis der Grobklassifizierung ist in Abbildung 6-17 dargestellt.

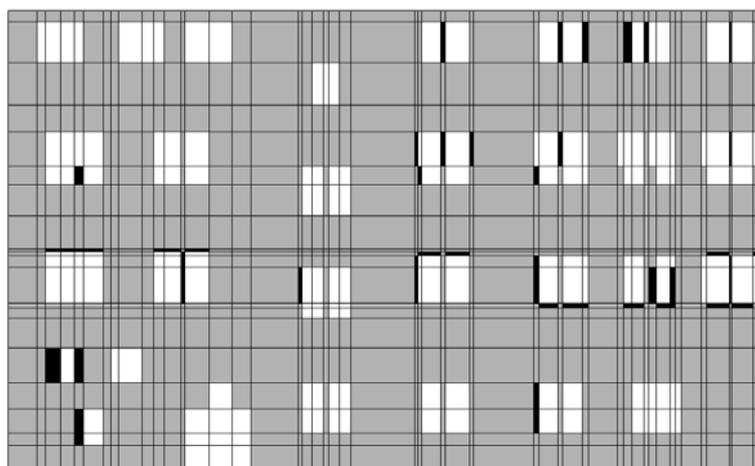


Abbildung 6-17: Zellenzerlegung nach der Grobklassifikation in Fassadenzellen (grau), Fensterzellen (weiß) und unsichere Zellen (schwarz).

6.3.5 Feinklassifizierung

Zellen, deren Rasterverhältnis keine eindeutige Klassifizierung mit den oben genannten Schwellwerten zulassen, sind oftmals sehr schmal und befinden sich nahe an den Fensterkanten. Für diese unsicheren Zellen wird mithilfe von heuristischen Regeln eine Feinklassifizierung durchgeführt. Durch die Betrachtung von sicher klassifizierten Zellen wird auf die Klassifizierung der noch unsicheren Zellen geschlossen. Dabei sind die heuristischen Regeln so formuliert, dass einfache Fenstergeometrien favorisiert werden und ein Ausgleich in der Lage und der Größe der Fenster geschieht:

1. Unsichere Zellen, die zwischen zwei sicheren Fensterzellen liegen, werden als Fassadenzellen klassifiziert. Dies lässt sich damit begründen, dass im Fassadenbereich der unsicheren Zelle tatsächlich Punkte gemessen wurden. Es ist jedoch anzunehmen, dass Fensterzellen, die zwischen zwei anderen Fensterzellen liegen, eine sehr niedrige Abdeckung mit Fassadenpunkten besitzen und deshalb sicher als Fensterzelle klassifiziert werden. Dieses Vorgehen hat beispielsweise zur Folge, dass statt einem großen Fenster zwei kleine Fenster rekonstruiert werden.
2. Als nächstes werden unsichere Zellen, die zwischen sicher klassifizierten Fenster- und Fassadenzellen liegen, in obere, untere, rechte und linke Diskontinuitäten unterschieden. Danach erfolgt eine Anpassung der Klassifizierung an die am nächsten gelegene, sichere Zelle mit derselben Konstellation. Dabei werden für obere und untere Diskontinuitäten nur Zellen in derselben Zeile und für rechte und linke Diskontinuitäten nur Zellen in derselben Spalte berücksichtigt.
3. Um geometrisch einfache Fensterobjekte zu erhalten, werden die noch immer unsicheren Zellen in den Fensterecken an die Nachbarzellen angepasst. Dieser letzte Schritt in der Feinklassifikation erzwingt eine konvexe Fensterform.

Nach der Klassifikation können die Fensterzellen aus dem Modell entfernt und die verbleibenden Fassadenzellen zur Gebäudefassade zusammengeklebt werden. Das Ergebnis der Feinklassifikation ist in Abbildung 6-18 zu sehen.

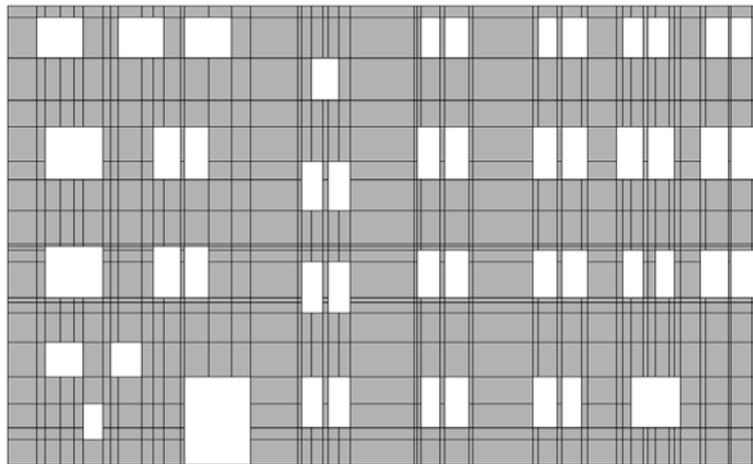


Abbildung 6-18: Zellenzerlegung nach der Feinklassifikation in Fassaden- (grau) und Fensterzellen (weiß).

6.3.6 Feinmodellierung mittels Hypothesentests

Die Fassadenmodellierung geschieht bisher durch Identifikation und Entfernen von Fensterzellen. Änderungen an der Zellenzerlegung wurden bisher nicht durchgeführt. Bei der Feinmodellierung mittels Hypothesentest werden nun gezielt Fassaden- und Fensterbereiche untersucht, um die Zellenzerlegung lokal an die gemessene Realität anzugleichen. Dabei werden Hypothesen über die Fassadenstruktur aufgestellt, um sie anschließend mithilfe der gemessenen Laserpunkte zu bestätigen oder ansonsten wieder zu verwerfen. Die Arbeitsweise unterscheidet sich von der bisherigen dahingegen, dass in der Punktwolke nicht mehr explizit nach Kanten gesucht wird, sondern eine Annahme über die lokale oder globale Struktur der Fassade getroffen wird. Aus der Annahme heraus wird die entsprechende Fassadengeometrie erzeugt und die neue Geometrie mit der Punktwolke verglichen. Als Qualitätskriterium kann beispielsweise die Summe der senkrechten

Abstände aller Punkte zur rekonstruierten Fassadenfläche dienen. Stellt sich heraus, dass die neue Fassadengeometrie die Punktwolke besser beschreibt, so ist die getroffene Annahme vermutlich richtig und die Fassadenstruktur wird beibehalten.

Zwei Strukturen werden bei der Feinmodellierung untersucht:

1. **Doppelfenster:** Wie bei der Feinklassifikation wird überprüft, ob ein großes Fenster durch zwei kleine, dicht nebeneinander liegende Fenster ersetzt werden kann. In diesem Fall existiert allerdings keine als unsicher klassifizierte, schmale Zelle, welche durch zwei Fensterzellen getrennt sind. Dies kann für Fassadenelemente geschehen, deren Ausdehnungen in der Größenordnung des verwendeten Suchradius liegen. In diesem Fall detektiert die Segmentierung nur eine statt zwei Kanten. Die große Fensterzelle wird daher für den Hypothesentest in zwei kleine Fensterzellen und eine schmale Fassadenzelle aufgeteilt. Diese Konfiguration wird mithilfe der ersten heuristischen Klassifikationsregel aus der Grobklassifizierung getestet und bei einer positiven Überprüfung beibehalten. Die Breite des schmalen Fassadenelements lässt sich nachträglich an die gemessenen Punkte in diesem Bereich anpassen.

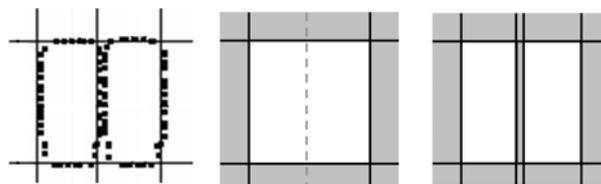


Abbildung 6-19: Segmentierungsergebnis für Fensterpunkte (links), daraus resultierende Zellenzerlegung vor (Mitte) und nach der Feinklassifizierung (rechts).

2. **Rundbögen:** Da durch die Zellenzerlegung mit Ebenen keine Rundbögen modelliert werden können, müssen diese nachträglich in das Modell integriert werden. Bei einem Fenster mit einem oberen Rundbogen werden die linke, rechte und untere Kante korrekt detektiert. Die obere Begrenzung wird jedoch bestenfalls im Bereich der Rundung gefunden oder durch eine andere horizontale Fensterkante definiert, im schlimmsten Fall fehlt diese jedoch komplett. Das Ergebnis ist eine Zelle, die im unteren Bereich eine sehr geringe und im oberen Bereich eine hohe Abdeckung mit Fassadenpunkten hat. Um den Rundbogen richtig zu setzen, muss zuerst der oberste Punkt der Bogenform gefunden werden. Dies geschieht, indem eine Ebene von der unteren Kante schrittweise hochgezogen wird, bis über der Ebene eine Abdeckung mit Fassadenpunkten von ca. 90% erreicht wird. Nun wird der Rundbogen so eingepasst, dass dieser an die obere Ebene und die beiden Seitenkanten angrenzt. Ist nun die Abdeckung mit Fassadenpunkten oberhalb des Rundbogens größer als 70% und unterhalb kleiner als 30%, so wird die Konfiguration beibehalten. Statt einem halbkreisförmigen Rundbogen können auch ovale Rundbögen modelliert werden. Dabei wird der vertikale Radius schrittweise verändert und der Rundbogen mit der besten Abdeckung mit Fassadenpunkten beibehalten.

Dieses sind nur zwei Beispiele für eine Feinmodellierung der Fenstergeometrien. Mit derselben Vorgehensweise sind aber auch weitere Maßnahmen denkbar. Man könnte Hypothesen über die Anzahl der Flügel, Existenz von Oberlicht oder das Sprossenbild aufstellen und diese gegen die gemessene Punktwolke testen. Die Fensterkonfiguration, die am Besten zu den Messpunkten passt, ist dann die wahrscheinlichste. Da sich die Fenster innerhalb derselben Fassade üblicherweise kaum unterscheiden, müssen nicht mal alle Fenster über eine ausreichende Punktabdeckung verfügen. Es reicht auch aus, wenn für jedes Fenster nur Teilbereiche gemessen wurden, da der Hypothesentest auf alle Fensterobjekte ausgedehnt werden kann. Die Konfiguration, die für alle Fenster mit derselben Größe und ähnlicher Lage am Besten passt, wird beibehalten.

Das vorgestellte Verfahren nutzt die Regelmäßigkeit der Struktur von Gebäudefassaden aus, um mittels vertikaler und horizontaler Schnitte eine Zellenzerlegung zu erzeugen. Aus dieser können Fenster und Türen abgeleitet werden, welche einheitliche Begrenzungslinien sowie Größe und Lage besitzen. Zwei Beispielergebnisse sind in Abbildung 6-20 und Abbildung 6-21 zu sehen.

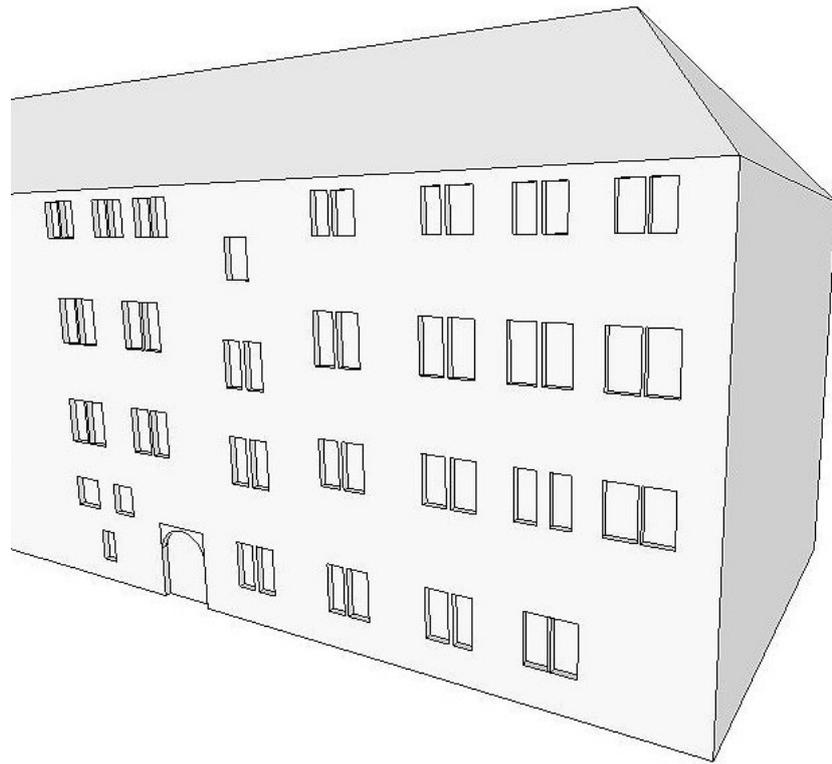


Abbildung 6-20: Ergebnis der Fassadenrekonstruktion.

Für den Ausbau des Verfahrens sollten weitere Fassadenelemente integriert werden. Insbesondere eine höhere Detaillierung der Fenster und die Hinzunahme von hervorstehenden Fassadenelementen wären denkbar (siehe Abbildung 6-21). Beispielsweise könnten die möglichen Anordnungen der verschiedenen Fensterelemente ähnlich wie in [Müller et al. 2006b] mittels Grammatiken ausgedrückt werden. Aus den parametrisierten Fenstergrundprimitiven (Flügel, Sprossen, Fenstersimse, etc.) sind dank der Grammatikregeln nur gültige Fenster ableitbar. Diese Grammatiken könnten auch gleich Generalisierungsregeln mit integrieren, mit denen die Fensterstruktur in eine größere Detaillierungsstufe umzuwandeln ist.

Die geometrische Modellierung mittels Zellenzerlegung ist ein mächtiges Werkzeug, ohne dabei die Komplexität und die damit verbundenen Probleme der Konstruktion mit Raumprimitiven zu erreichen. Trotzdem kann zeitgleich mit der Rekonstruktion ein einfacher CSG-Baum für die Fassade aufgestellt werden, wie er z.B. von Thiemann und Sester [2004] benötigt wird, um eine Fassadengeneralisierung durchzuführen. Eine vorherige Aufteilung des Gebäudes in seine Bestandteile wäre somit unnötig um die dort vorgeschlagenen Generalisierungsmaßnahmen Weglassen, Verdrängen, Betonen, Zusammenfassen und Typifizieren auf die Fassade anzuwenden.

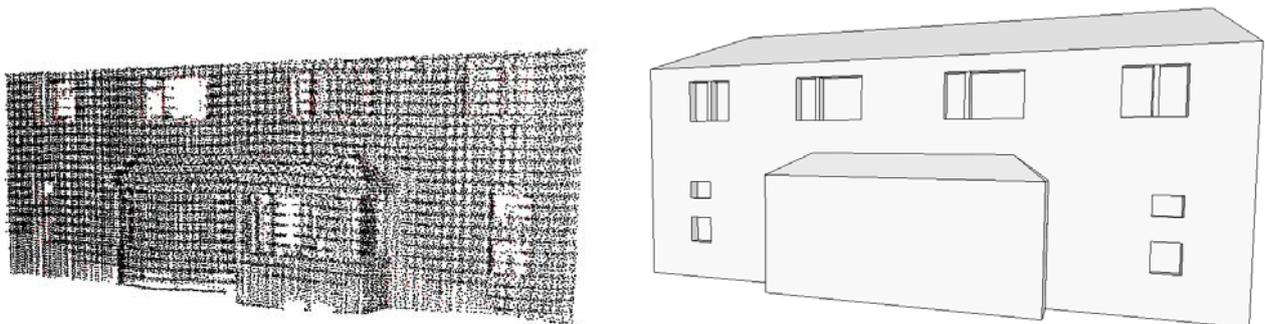


Abbildung 6-21: Laserpunktwolke und rekonstruierte Ergebnismodell mit Fassadenstruktur.

7 Diskussion und Ausblick

Mit der vorliegenden Arbeit werden automatische Verfahren und Methoden zur maßstabsabhängigen Erzeugung von 3D-Gebäudemodellen aufgezeigt. Unterschiedliche Detailstufen werden in vielfältigen Anwendungsgebieten benötigt, um kartenverwandte 3D-Darstellungen von urbanen Landschaften zu generieren. Eine flächendeckende Rekonstruktion solcher Modelle ist bisher jedoch nur aus luftgestützt erfassten Daten möglich. Der bisher insbesondere für Städte und Kommunen erfasste Bestand besteht folglich aus Modellen, die sehr detaillierte Grundrisse und Dachstrukturen besitzen, aber nur flache Fassadenelemente ohne Fenster, Türen, Balkone, etc. aufweisen. Damit Daten nicht für jeden nachgefragten Maßstab neu erfasst werden müssen, sind Transformationsverfahren notwendig, um die 3D-Gebäudemodelle bei Bedarf zu generalisieren oder die Fassaden bildlich und geometrisch auszugestalten. Die vorgestellten Verfahren vereinfachen dabei Einzelgebäude geometrisch bis hin zu Blockmodellen, oder werten diese zu Architekturmodellen mit detaillierten Fassaden- und Dachstrukturen mit hoch aufgelösten Texturen auf. Mit Ausnahme der Innenraummodelle können damit alle Detailstufen von 3D-Gebäudemodellen erzeugt werden, wie sie auch im CityGML Standard aufgeführt sind.

Es werden zwei komplementäre Ansätze zur geometrischen Vereinfachung beschrieben. Das erste Verfahren greift die Ideen aus der klassischen 2D-Grundrissgeneralisierung auf, um diese auf die dritte Dimension zu erweitern. Mittels expliziter Regeln werden iterativ Merkmale des polygonalen 3D-Gebäudemodells detektiert, klassifiziert und anschließend entsprechend ihres Typs aus dem Modell entfernt. Dieser Schritt ähnelt sehr denen der Generalisierungsansätze, die auf Maßstabsräumen oder der allgemeinen Segmentierung basieren. Die jeweiligen Regelwerke zur Merkmalsbehandlung sind somit grundsätzlich austauschbar und wären sogar in Kombination alternierend anwendbar. Diese Interpretation der Gebäudestruktur ist äußerst wichtig für die iterativen Generalisierungsalgorithmen, damit die Merkmale typgerecht behandelt werden können und dabei die für Gebäude wichtigen geometrischen Relationen erhalten bleiben. Dadurch unterscheidet sich die 3D-Gebäudegeneralisierung von der Simplifizierung allgemeiner Oberflächen, der die Interpretation und objektspezifische Vereinfachung fehlen. Die komplexen Generalisierungsoperatoren werden im vorgestellten Verfahren als Kombination der elementaren Operatoren für Modelle in Randbeschreibung formuliert. Sie vereinfachen 3D-Gebäudemodelle unter Bewahrung von Koplanarität, Parallelität und Rechtwinkligkeit der Gebäudeflächen.

Die Einhaltung der geometrischen Relationen wird auch durch die Einführung eines Relationsgraphen sichergestellt, der die Relationen verwaltet und anhand dessen die Gültigkeit jeder Operation überprüft werden kann. Eine Generalisierungsoperation muss das Gebäudemodell wieder in einen konsistenten Zustand bringen. Ansonsten ist die Operation ungültig und darf nicht ausgeführt werden bzw. muss wieder rückgängig gemacht werden. So ein Kontrollmechanismus könnte auch für alternative Verfahren nützlich sein, die auf Methoden der künstlichen Intelligenz aufbauen. Beispielsweise könnte ein neuronales Netz lernen, die elementaren Operationen solange zu kombinieren, bis gültige Operationen entstehen. Dabei ist nicht nur eine Überprüfung der Vereinfachungsoperation denkbar, sondern auch des Generalisierungsvorgangs Vergrößern.

Nach der Vereinfachung erfolgt eine Einpassung der neuen Gebäudeform an die ursprüngliche Geometrie. Auf diese Weise werden Verzerrungen und Verschiebungen ausgeglichen, die sich durch die Merkmalsentfernung ergeben können. Dieser Vorgang ist im Prinzip immer anwendbar, wenn ein vereinfachtes 3D-Gebäudemodell in Randbeschreibung an die ursprüngliche Form angepasst werden soll. Beispielsweise könnte man die Formeinpassung auch nach der Generalisierung aus Kapitel 5 vornehmen. Hierzu müssen jedoch die geometrischen Relationen der Gebäudeflächen aufgestellt und die Originalpunkte den Flächen des generalisierten Modells zugeordnet werden.

Die aus Experimenten gewonnenen Ergebnisse zeigen, dass die von unten nach oben gerichtete (bottom up) Vorgehensweise geeignet ist, um Strukturelemente unter Beachtung von Generalisierungsregeln zu eliminieren. Für jeden Merkmalstyp muss jedoch ein eigenes Regelwerk erstellt werden, wobei noch keine Erkenntnisse vorliegen, wie viele Regeln insgesamt benötigt werden und wie komplex diese sein müssen, um alle Merkmale behandeln zu können. Hier könnten vielleicht die schon angesprochenen allgemeinen Operatoren

eine Lösung sein. Dies ist auch abhängig von Rekonstruktionsmethode, die für die Komplexität des zu generalisierenden 3D-Gebäudemodells verantwortlich ist. Zu den Vorzügen des Verfahrens gehören, dass Texturdaten aufgrund der eingesetzten Operatoren nahezu unverändert beibehalten werden können, und dass die iterative Vorgehensweise eine progressive Speicherung aller Zwischenergebnisse ermöglicht. Letztere Eigenschaft ist vor allem für Echtzeitanwendungen wichtig, die detaillierte 3D-Stadtmodelle über schmalbandige Kommunikationskanäle empfangen. So können zunächst die groben Modelle übertragen und präsentiert werden, um anschließend schrittweise weitere Details hinzuzufügen.

Die Vereinfachung im zweiten Generalisierungsverfahren erfolgt mit einer entgegengesetzten Herangehensweise. Das 3D-Gebäudemodell wird nicht mehr iterativ vereinfacht. Stattdessen wird in einem von oben nach unten gerichteten (top down) Ansatz ein neues 3D-Modell generiert, welches die globale Gebäudeform widerspiegelt. Dies erfolgt in einem zweigeteilten Prozess. Zuerst wird eine Zellenzerlegung aus den vertikalen Fassadenflächen erzeugt, so dass eine Art generalisierter Grundriss von nicht überlappenden Polygonelementen entsteht. Die Schnittgrenzen ergeben sich aus approximierenden Ebenen, die über die Anwendung von Puffern gebildet werden. Anschließend wird für jede Grundrisszelle ein Dachtyp ermittelt und die Zelle entsprechend um die dritte Dimension erweitert. Dies kann wieder durch Pufferbildung erfolgen, wodurch sich sehr allgemeine Formen finden lassen. Der Nachteil ist, dass manchmal sehr kleine Zellen entstehen, die nicht mehr korrekt zur generalisierten Gebäudeform zugeordnet werden. Dadurch entstehen unnötige Details, die mittels spezieller Optimierungsregeln identifiziert und behandelt werden müssen. Es wird gezeigt, dass dieser Ansatz in der Lage ist, 3D-Gebäudemodelle unterschiedlicher Komplexität zu generalisieren und dabei die geometrischen Relationen nicht zu verletzen. Selbst komplexe Dachstrukturen mit mehreren Dachebenen und Dachformen bleiben nach der Vereinfachung erhalten. Somit wird die globale Gebäudeform korrekt gefunden, was insbesondere für Landmarken und für deren Wiedererkennung von enormer Wichtigkeit ist.

Weiterhin wird ein alternativer Ansatz vorgestellt, bei dem die dreidimensionalen Zelltypen aus einer Menge vordefinierter Dachformen gewählt werden. Um die Auswahl möglicher Dachtypen einzuschränken und dadurch einen glatten Übergang zwischen den Zellen zu garantieren, werden zusätzliche Konsistenzregeln formuliert. Es findet somit eine Interpretation der Dachform statt, wodurch das Generalisierungsverfahren einer 3D-Gebäuderekonstruktion ähnelt. Der Unterschied ist, dass im Gegensatz zur Rekonstruktion die Interpretation bei der Generalisierung auf vorinterpretierten Polygondaten erfolgt. Damit sind große, dominante Flächen und klare Bruchkanten vorhanden und die Ausrichtung der Flächen kann bei der Auswahl des Dachtyps mit berücksichtigt werden. Es wird außerdem gezeigt, dass durch das Wissen über die Dachform auch eine gezielte Typifizierung möglich ist, indem beispielsweise die Anzahl der Dachkanten bei Paralleldächern reduziert wird. Des Weiteren ist eine Erweiterung des Generalisierungsansatzes um zusätzliche elementarer Vorgänge wie Vergrößern, Zusammenfassen, Bewerten, etc. basierend auf der Interpretation denkbar.

Die Generalisierung von 3D-Gebäudemodellen steckt noch immer in den Kinderschuhen. Dies ist nicht weiter verwunderlich, da auch die Erfassung und Visualisierung von 3D-Stadtmodellen erst in jüngster Zeit dem Forschungsstadium entsprungen ist und sich aus dem Themengebiet kommerzielle Dienstleistungen und Produkte entwickelt haben. Bedenkt man die Anzahl der unterschiedlichen elementaren Vorgänge kartographischer Generalisierung, so wurde mit der Vereinfachung von Einzelgebäuden nur ein geringer Bereich des Themengebiets bearbeitet. Neben wenigen Beiträgen zur Aggregation, Klassifizierung und Typifizierung ist das Forschungsgebiet ansonsten unberührt. Aber auch die Vereinfachung ist noch lange nicht erschöpfend untersucht worden. Bisherige Verfahren sind nicht in der Lage, Gebäudestrukturen mit gekrümmter oder kreisförmiger Grundfläche geeignet zu generalisieren. Doch gerade markante Gebäude wie Kirchen, Burgen, Schlösser, etc. besitzen oftmals runde Türme, die für das charakteristische Erscheinungsbild der Objekte selbst und dem gesamten Stadtbild mit verantwortlich sind. Daher ist es wichtig, dass bei der Vereinfachung der entsprechenden Gebäudeteile deren relevanten, geometrischen Eigenschaften zwingend erhalten bleiben.

Da runde Gebäudeelemente in Konflikt stehen zu den geometrischen Relationen Koplanarität, Parallelität und Rechtwinkligkeit, kann eine Erweiterung des auf der Zellenzerlegung basierenden Generalisierungsansatzes am besten durch einen vorgeschalteten Prozess realisiert werden. Beispielsweise könnte für kreisförmig angeordnete, vertikale Flächen ein Kreismittelpunkt ermittelt werden, aus dem sich ein parametrisiertes Turmprimitiv mit Kuppel-, Kegel-, Zwiebel- oder Glockendach ableiten lässt. Nach einem erfolgreichen Hypothesentest, der das Vorhandensein des Elements mit der ermittelten Parametrisierung bestätigt, werden der Grundriss und die Dachstruktur entsprechend des Primitivtyps vereinfacht und nach der Generalisierung

des Restgebäudes mittels Boolescher Vereinigungsoperation dem Gebäude hinzugefügt. Damit basiert der Generalisierungsansatz nicht mehr allein auf der Zellenzerlegung, sondern enthält auch Operatoren und die damit verbundenen Implementierungsschwierigkeiten der Konstruktion mit Raumprimitiven. Das restliche Gebäude kann wie in Kapitel 5 beschrieben, geometrisch vereinfacht werden. Um jedoch die Ableitung der approximierenden Schnittebenen nicht negativ durch die runden Elemente zu beeinträchtigen, müssten lediglich die entsprechenden Flächen temporär für diesen Generalisierungsschritt aus dem Modell entfernt werden.

Existierende Verfahren zur geometrischen Vereinfachung betrachten bisher isoliert die zu generalisierenden 3D-Gebäudemodelle. Da sich durch den Vorgang die Grundrisse häufig verändern, können bei kartenverwandten 3D-Darstellungen von urbanen Gebieten räumliche Konflikte zu benachbarten Gebäuden entstehen. Das Problem wird zusätzlich verschärft, wenn Landmarken vergrößert präsentiert werden sollen, um eine Betonung der relevanten Bauwerke zu erzielen. In Zukunft wird die Generalisierung daher auf ganze Straßenzüge oder Wohnblöcke ausgeweitet werden müssen. In diesem Zusammenhang ist die Entwicklung weiterer elementarer Generalisierungsvorgänge wie beispielsweise Aggregation, Bewerten und Selektion notwendig. In diesem Zusammenhang liefert die 2D-Grundrisszerlegung eine gute Ausgangsbasis für eine Aggregation nahe gelegener Gebäude, da dort die Geometrien der Zwischenräume als Zellen gegeben sind. Hier sind Regeln auszuarbeiten um die relevanten Zellen zu identifizieren.

Im letzten Teil der Arbeit wird die Ausgestaltung von Fassaden mittels terrestrisch erfasster Bild- und Laserdaten behandelt. Basierend auf der manuellen Texturierung von Fassaden wird ein automatischer Vorgang zur bildbasierten Ausgestaltung diskutiert. Es wird gezeigt, dass sich unter Ausnutzung spezieller Grafikhardware eine effiziente Prozesskette aufbauen lässt, die aus orientierten Bildern ohne Benutzerinteraktion texturierte 3D-Gebäudemodelle generiert. Dabei können Verdeckungen sowohl durch das Objekt selbst als auch durch Fremdobjekte erkannt und durch Fusion von Bildfolgen retuschiert werden. Einsetzbar wäre ein solches System z.B. für die Mobile Erfassung von 3D-Stadtmodellen. In diesem Bereich werden in den letzten Jahren zunehmend Bild- und Laserdaten von Fahrzeugen aus aufgenommen.

Für terrestrische Laserdaten wird der Ansatz der Zellenzerlegung auf die Rekonstruktion von Fassaden angewendet. Es wird ein Verfahren vorgestellt, welches mittels Segmentierung und einfach gestalteter Regeln Fenster und Türen in der Punktwolke detektiert und ausmodelliert. Beide Objekttypen sind dadurch gekennzeichnet, dass sich die gemessenen Punkte hinter der Fassadenebene befinden. In zukünftigen Arbeiten wird man daher untersuchen müssen, wie man hervorstehende Fassadenstrukturen modellieren kann. Insbesondere Regeln für beliebig strukturierte Erker und Ausluchten werden von Interesse sein, da diese Fassadenelemente das charakteristische Erscheinungsbild von Gebäuden prägen. Da durch die Rekonstruktion semantische Informationen über die Fassade abgeleitet werden, ist eine Fassadengeneralisierung denkbar. Durch den 2.5D-Charakter von Einzelfassaden können hier auch die elementaren Generalisierungsvorgänge, Verdrängen und Klassifizieren, eine größere Rolle spielen.

8 Literaturverzeichnis

- Albert, J., Bachmann, M., Hellmeier, A. [2003]. *Zielgruppen und Anwendungen für Digitale Stadtmodelle und Digitale Geländemodelle* – Erhebung im Rahmen der Arbeitsgruppe „Anwendungen und Zielgruppen“ der SIG3D im Rahmen der Initiative GDI-NRW.
- Alegre, F. und Dallaert, F. [2004], *A Probabilistic Approach to the Semantic Interpretation of Building Facades*. In: International Workshop on Vision Techniques Applied to the Rehabilitation of City Centres, S. 1-12.
- Alliez, P., Ucelli, G., Gotsman, C., Attene, M. [2005], *Recent Advances in Remeshing of Surfaces*. Technical Report. AIM@Shape Nextwork of Excellence.
- Ameri, B. [2000], *Automatic Recognition and 3D Reconstruction of Buildings through Computer Vision and Digital Photogrammetry*. Deutsche Geodätische Kommission, Reihe C, Nr. 526, München.
- Anders, K.-H. [2005], *Level of Detail Generation of 3D Building Groups by Aggregation and Typification*. In: Proceedings of the XXII International Cartographic Conference. La Coruna, Spain.
- Bajaj, C. und Kim, M.-S. [1988], *Generation of Configuration Space Obstacles: The Case of a Moving Sphere*. IEEE Journal of Robotics and Automation, Vol. 4 (1), S. 94-99.
- Baltsavias, E. Stallmann, D., Mason, S.O. [1995], *Use of DTMs/DSMs and Orthoimages to Support Building Extraction*. In: Grün, A., Kübler, O., Agouris, P. (Editors), Automatic Extraction of Man Made Objects from Aerial and Space Images, Birkhäuser Verlag, Basel, S. 199-210.
- Baltsavias, E. und Mason, S. [1997], *Automated Shack Reconstruction using Integration of Cues in Object Space*. In: Proceedings of the ISPRS Commission III / IV Workshop, Vol. 32-3-4 W2, S. 96-105.
- Bauer, W. und Mohl, H.-U. [2004], *Das 3D-Stadtmodell der Landeshauptstadt Stuttgart*. In: Coors, V., Zipf, A. (Editors), 3D-Geoinformationssysteme, Wichmann Verlag.
- Baumgart, B. [1974], *Geometric Modeling for Computer Vision*, Ph.D. thesis, Stanford University, 1974.
- Bildstein, F. [2005], *3D City Models for Simulation & Training – Requirements on Next Generation 3D City Models*. In: Proceedings of the ISPRS Workshop on Next Generation City Models, Bonn.
- Blythe, D. [2006], *The Direct3D 10 System*, In: ACM Transactions on Graphics, Volume 25 Number 3, Proceedings of the SIGGRAPH 2006, Boston, MA, S. 724-734.
- Böhm, J. [2004], *Multi-Image Fusion for Occlusion-Free Façade Texturing*. In: International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 35, ISPRS, Istanbul.
- Böhm, J. und Haala, N. [2005], *Efficient Integration of Aerial and Terrestrial Laser Data for Virtual City Modeling using Lasermaps*. In: Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 36, Part 3/W19. Proceedings of the ISPRS Workshop on Laser Scanning, Enschede, Niederlande, S. 192-197.
- Botsch, M., und Kobbelt, L. [2001], *Resampling Feature and Blend Regions in Polygonal Meshes for Surface Anti-Aliasing*. In: Proceedings of Eurographics 2001, Manchester, UK.
- Brenner, C. [2000], *Dreidimensionale Gebäuderekonstruktion aus digitalen Oberflächenmodellen und Grundrissen*. Dissertation. Deutsche Geodätische Kommission, Reihe C, Nr. 520.
- Brenner, C. [2004], *Modelling 3D Objects Using Weak CSG Primitives*, In: International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 35, ISPRS, Istanbul.
- Brenner, C. [2005], *Building Reconstruction from Images and Laser Scanning*, In: International Journal of Applied Earth Observation and Geoinformation, Theme Issue on “Data Quality in Earth Observation Techniques”, 6 (3-4), Elsevier, S. 187-198.

- Brenner, C. und Haala, N. [1998], *Rapid Acquisition of Virtual Reality City Models from Multiple Data Sources*. In: Chikatsu, H., Shimizu, E. (Editors), *International Archives of Photogrammetry & Remote Sensing*, Vol. 32, Part 5, S. 323-330.
- Brenner, C. und Sester, M. [2003], *Continuous Generalization for Small Mobile Displays*. Working Paper of the International Workshop on Next Generation Geospatial Information, Cambridge (Boston), USA.
- Brown, D.C. [1971]. *Close-Range Camera Calibration*. In: *Photogrammetric Engineering*, 37 (8), S. 855-866.
- Brunn, A. [1998], *Techniques for Automatic Building Extraction*. In: *Third Course in Digital Photogrammetry*, Institut für Photogrammetrie, Universität Bonn und Landesvermessungsamt Nordrhein-Westfalen.
- Buchholz, H., Döller, J., Nienhaus, M., Kirsch, F. [2005], *Real-Time Non-Photorealistic Rendering of 3D City Models*. In: *Proceedings of the 1st International Workshop on Next Generation 3D City Models*, Bonn.
- Chen, L.C., Teo, T.A., Shao, Y.C., Lai, Y.C., Rau, J.Y. [2004], *Fusion of LIDAR Data and Optical Imagery for Building Modeling*, In: *International Archives of Photogrammetry and Remote Sensing*, Vol. 35, B4, S. 732-737.
- Cignoni, P., Montani, C., Rocchini, C., Scopigno, R. [1998], *Zeta: A Resolution Modeling System*. In: *GMIP: Graphical Models and Image Processing* 60, 5, S. 305-329.
- Cignoni, P., Constanza, D., Montani, C., Rocchini, C., Scopigno, R. [2000], *Simplification of Tetrahedral Meshes with Accurate Error Evaluation*. In: *IEEE Visualization 2000*, S. 85-92.
- Cohen, J., Olano, M., Manocha, D. [1998], *Appearance-Preserving Simplification*. In: *Proceedings of ACM SIGGRAPH 1998*, Orlando, Florida, USA, S. 115-122.
- Coorg, S. und Teller, S. [1999], *Extracting Textured Vertical Facades from Controlled Close-Range Imagery*. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, S. 625-632.
- Coors, V. [2001], *Feature-Preserving Simplification in Web-Based 3D-GIS*. In: *Proceedings of the 1st International Symposium on Smart Graphics*. Hawthorne, NY, USA, S. 22-28.
- Cox, S., Daisy, P., Lake, R., Portele, C., Whiteside, A. [2004], *OpenGIS Geography Markup Language (GML3.1), Implementation Specification Version 3.1.0*, Recommendation Paper, OGC Doc, No. 03-105r1.
- Debevec, P.E., Taylor, C.J., Malik, J. [1996], *Modeling and Rendering Architecture from Photographs: A Hybrid Geometry- and Image Based Approach*. In: *Proceedings of SIGGRAPH 1996*, S. 11-20.
- De Floriani, L. [1989], *Feature Extraction from Boundary Models of Three-Dimensional Objects*. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 11 (8), S. 785-798.
- Dick, A., Torr, P., Cipolla, R. [2004], *Modelling and Interpretation of Architecture from Several Images*. In: *International Journal of Computer Vision*, Vol 60 (2), S. 111-134.
- Döllner, J. und Walther, M. [2003], *Real-Time Expressive Rendering of City Models*. In: *Proceedings of the IEEE 2003 Information Visualization, 7th International Conference on Information Visualization*, S. 245-250, London.
- Dorffner, L. und Forkert, G. [1998], *Generation and Visualization of 3D Photo-Models using Hybrid Block Adjustment with Assumptions on the Object Shape*. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 53 (6), S. 369-378.
- Douglas, D.H. und Peucker, T.K. [1973], *Algorithms for the Reduction of the Number of Points required to Represent a Line or its Caricature*. In: *The Canadian Cartographer*, 10 (2), S. 112-122.
- Duchene, C., Bard, S., Barillot, X., Ruas, A., Trevisan, J., Holzapfel, F. [2003], *Quantitative and Qualitative Description of Building Orientation*, In: *Working Paper of the ICA Workshop*, Paris, Frankreich, 2003

- Ehrenfels, C. von [1890], *Über Gestaltqualitäten*. In: Vierteljahresschrift für wissenschaftliche Philosophie 4, S. 249-292.
- El-Hakim, S.F. [2002], *Semi-Automatic 3D Reconstruction of Occluded and Unmarked Surfaces from Widely Separated Views*. In: Proceedings of the ISPRS Symposium, Corfu, Griechenland, S. 143-148.
- El-Hakim, S.F. [2006], *A Sequential Approach to Capture Fine Geometric Details from Images*. In: Proceedings of the ISPRS Commission V Symposium – Image Engineering and Vision Metrology, Vol. 36, Part 5, Dresden, S. 97-102.
- El-Sana, J. und Varshney, A. [1998], *Topology Simplification for Polygonal Virtual Environments*. In: IEEE Transactions on Visualization and Computer Graphics, Vol. 4 (2), S. 133-144.
- Faugeras, O. und Keriven, R. [1998], *Complete Dense Stereovision using Level Set Methods*. In: Proceedings of the 5th European Conference on Computer Vision (ECCV '98), Volume I, S. 379-393.
- Fernando, R. und Kilgard, M. [2003], *The Cg Tutorial*, Addison Wesley.
- Fischler, M., Bolles, R. [1981], *Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography*. In: Communication of the ACM, Vol. 24 (6), S. 381-395.
- Förstner, W. und Gülch, E. [1987], *A Fast Operator for Detection and Precise Location of Distinct Points, Corners and Centers of Circular Features*. In: Proceedings of the ISPRS Intercommission Workshop on Fast Processing of Photogrammetric Data.
- Förstner, W. [1999], *3D-City Models: Automatic and Semiautomatic Acquisition Methods*. In: Fritsch, Spiller (Editors), Photogrammetric Week '99, Wichmann Verlag, Heidelberg, S. 291-303.
- Foley, J., van Dam, A., Feiner, S., Hughes, J. [1990], *Computer Graphics: Principles and Practice (2nd Edition)*, Addison-Wesley.
- Forberg, A. und Mayer, H. [2002], *Generalization of 3D Building Data Based on Scale-Spaces*. In: Proceedings of the Joint International Symposium on Geospatial Theory, Processing and Applications, Ottawa, Kanada, S. 225-230.
- Forberg, A. und Mayer, H. [2003], *Squaring and Scale-Space Based Generalization of 3D Building Data*. In: Working Paper of the ICA Workshop, Paris, Frankreich, 2003.
- Forberg, A. [2004], *Generalization of 3D Building Data Based on a Scale-Space Approach*. In: Proceedings of the XXth Congress of the ISPRS, Vol. 35, Part B, Istanbul, Türkei, 2004.
- Forberg, A. [2005], *Generalisierung dreidimensionaler Gebäudedaten auf der Basis von Maßstabsräumen*, Elektronische Dissertation, Universität der Bundeswehr München, Fakultät für Bauingenieur- und Vermessungswesen.
- Fraser, C.S. [1997], *Digital Camera Self-Calibration*. In: ISPRS Journal of Photogrammetry and Remote Sensing, 52, S. 149-159.
- Früh, C. und Zakhor, A. [2003] *Constructing 3D City Models by Merging Ground-Based and Airborne Views*, In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '03) – Volume 2, S. 562-572.
- Früh, C., Jain, S., Zakhor, A. [2005] *Data Processing Algorithms for Generating Textured 3D Building Façade Meshes from Laser Scans and Camera Images*. In: International Journal on Computer Vision, Vol. 61 (2), S. 159-184.
- Fuchs, H., Kedem, Z.M., Naylor, B.F. [1980], *On Visible Surface Generation by A Priory Tree Structures*, In: Proceedings of SIGGRAPH 1980, Seattle, WA, USA, S. 124-133.
- Garland, M. und Heckbert, P. [1997], *Surface Simplification using Quadric Error Metrics*, In: Proceedings of SIGGRAPH 97, Los Angeles, USA, S. 206-216.
- Garland, M. und Heckbert, P. [1998], *Simplifying Surfaces with Color and Texture using Quadric Error Metrics*, In: Proceedings of IEEE Visualization 98, IEEE CS Press, Los Alamitos, USA, S. 263-270.

- Garland, M., [1999], *Quadric-Base Polygonal Surface Simplification*. Ph.D. Dissertation, Computer Science Department, CMU-CS-99-105.
- Gavankar, P. und Henderson, M.R. [1990], *Graph-Based Extraction of Protrusions and Depressions from Boundary Representations*. In: Computer Aided Design 22 (7), S. 442-450.
- Gray, K. [2003], *The Microsoft DirectX 9 Programmable Graphics Pipeline*. Microsoft Press, USA.
- Gröger, G., Kolbe T.H., Plümer, L. [2004], *Zur Konsistenz bei der Visualisierung multiskaliger 3D-Stadtmodelle*. In: Mitteilungen des Bundesamtes für Kartographie und Geodäsie, Band 31, September 2004, Verlag des Bundesamtes für Kartographie und Geodäsie.
- Gröger, G., Kolbe, T.H., Czerwinski, A. [2006], *City Geography Markup Language*. Approved Discussion Paper of the Open Geospatial Consortium, Inc. <http://www.opengeospatial.org/standards/dp>
- Grossmann, E. und Santors-Victor, J. [2005], *Least-Squares 3D Reconstruction from one or more Views and Geometric Clues*. In: Computer Vision and Image Understanding, Vol. 99 (2), S. 151-174.
- Grün, A. und Wang, X. [1998], *CC-Modeler: A Topology Generator for 3-D City Models*. In: Fritsch, D., Englich, M., Sester, M. (Editors), International Archives of Photogrammetry and Remote Sensing, Vol. 32, Part 4, Stuttgart, S. 188-196.
- Grün, A. und Wang, X. [2001], *News from CyberCity-Modeler*, In: Automatic Extraction of Man-Made Objects from Aerial and Space Images (III), Swets & Zeitlinger B.V., Lisse, Niederlande, S. 93-101.
- Grünreich, D. [1985], *Ein Vorschlag zum Aufbau einer großmaßstäbigen, topographisch-kartographischen Datenbank unter besonderer Berücksichtigung der Grundrissdatei des ALK-Systems*. In: Nachrichten aus dem Karten- und Vermessungswesen, Band 1, Nr. 95, S. 53-62.
- Gülch, E., Müller, H., Läbe, T. [1998], *Integration of Automatic Processes into Semi-Automatic Building Extraction*. In: International Archives of Photogrammetry & Remote Sensing, Vol. 32, Part 3, München.
- Gülch, E., Müller, H., Läbe, T., Ragia, L. [1999], *On the Performance of Semi-Automatic Building Extraction*. In: Proceedings of the ISPRS Commission III Symposium, Columbus, Ohio, USA.
- Guibas, L. and Stolffy, J. [1985], *Primitives for the Manipulation of General Subdivisions and the Computation of Voronoi Diagrams*. In: Transactions on Graphics, Vol. 4, No. 2, S. 74-123.
- Haala, N. [1996], *Gebäuderekonstruktion durch Kombination von Bild- und Höhendaten*, Dissertation, Universität Stuttgart, Institut für Photogrammetrie, Deutsche Geodätische Kommission, C 460.
- Haala, N., Anders K.-H. [1997], *Acquisition of 3D Urban Models by Analysis of Aerial Images, Digital Surface Models and Existing 2D Building Information*, In: Proceedings of SPIE Conference on Integrating Photogrammetric Techniques with Scene Analysis und Machine Vision III, S. 212-222.
- Haala, N., Brenner, C., Anders, K.-H. [1998], *Urban GIS from Laser Altimeter and 2D Map Data*. In: Proceedings of the ISPRS Commission III Symposium, Vol. XXXII-3/1 of International Archives of Photogrammetry and Remote Sensing, Columbus, Ohio, S.339-346.
- Hake, G., Grünreich, D., Meng, L. [2002], *Kartographie – Visualisierung raum-zeitlicher Informationen*, de Gruyter, Berlin.
- Harrie, L.E. [1999], *The Constraint Method for Solving Spatial Conflicts in Cartographic Generalisation*. In: Cartography and Geographic Information Systems.
- Harrie, L., Sarjakoski, T., Lehto, L. [2002], *A Variable-Scale Map for Small-Display Cartography*. In: Proceedings of the Joint International Symposium on Geospatial Theory, Processing and Applications, Ottawa, Kanada.
- He, T., Hong, L., Kaufman, A., Varshney, A., Wang, S. [1995], *Voxel-Based Object Simplification*. In: Proceedings of Visualization 1995, IEEE CS Press, Los Alamitos, USA, S. 296-303.
- Heckbert, P.S. und Garland, M. [1997], *Survey of Polygonal Simplification Algorithms*. In: Multiresolution Surface Modeling Course, SIGGRAPH '97, Los Angeles, USA.

- Henricsson, O. [1996], *Analysis of Image Structures using Color Attributes and Similarity Relations*. Dissertation, Institut für Geodäsie und Photogrammetrie, ETH Zürich, Mitteilungen Nr. 59.
- Henricsson, O., Bignone, F., Willihn, W., Ade, F., Kuebler, O., Baltsavias, E., Mason, S.O., Grün, A. [1996], *Project AMOBE: Strategies, Current Status and Future Work*. In: International Archives of Photogrammetry and Remote Sensing, Vol. 31, Part B3, S. 321-330.
- Hilton, A. [2005], *Scene Modelling from Sparse 3D Data*. In: Image and Vision Computing, Vol. 23 (10), S. 900-920.
- Hofmann, A. [2005], *An Approach to 3D Building Model Reconstruction from Airborne Laser Scanner Data using Parameter Space Analysis and Fusion of Primitives*. Dissertation. Institut für Photogrammetrie und Fernerkundung, TU Dresden.
- Hoffmann, C. [1989], *Solid and Geometric Modeling*. Morgan Kaufmann Publishers, San Mateo, CA, USA.
- Hoppe, H. [1996], *Progressive Meshes*, In: Proceedings of SIGGRAPH '96, New Orleans, LA, USA, S. 99-108.
- Hoppe, H. [1999], *New Quadric Metric for Simplifying Meshes with Appearance Attributes*, In: Proceedings of Visualization 1999, San Francisco, USA, S. 59-66.
- Hoppe, H., DeRose, T., Duchamp, T., McDonald, J. Stuetzle, W. [1993], *Mesh Optimization*. In: Proceedings of ACM SIGGRAPH 1993, S.19-26.
- Jang, J., Ribarsky, W., Shaw, C., Wonka, P. [2003], *Appearance-Preserving View-Dependent Visualization*. In: Proceedings IEEE Visualization, S. 473-480.
- Jang, J., Wonka, P., Ribarsky, W., Shaw, C.D. [2005], *Punctuated Simplification of Man-Made Objects*. In: The Visual Computer. International Journal of Computer Graphics, Vol. 22 (2), S. 136-145.
- Kaartinen, H., Hyypä, J., Gülch, E., Vosselman, G., Hyypä, H., Matikainen, L., Hofmann, A.D., Mäder, U., Persson, Å., Söderman, U., Elmqvist, M., Ruiz, A., Dragoja, M., Flamanc, D., Maillet, G., Kersten, T., Carl, J., Hau, R., Wild, E., Frederiksen, L., Holmgaard, J., Vester, K. [2005], *Accuracy of 3D City Models: EuroSDR Comparison*. In: Proceedings of the ISPRS WG III/3, III/4, V/3 Workshop "Laser Scanning 2005", Enschede, Niederlande.
- Kada, M., [2002], *Automatic Generalisation of 3D Building Models*. In: Proceedings of the Joint International Symposium on Geospatial Theory, Processing and Applications, Ottawa, Canada.
- Kada, M., [2005], *3D Building Generalisation*. In: Proceedings of the 22th International Cartographic Conference, La Coruna, Spanien.
- Kada, M., Roettger, S., Weiss, K., Ertl, Th., Fritsch, D. [2003], *Real-Time Visualisation of Urban Landscapes Using Open Source Software*, In: Proceedings of the 24th Asian Conference on Remote Sensing & 2003 International Symposium on Remote Sensing, Busan, Korea.
- Kho, Y.S. und Garland, M. [2003], *User-Guided Simplification*. In: Proceedings of the ACM Symposium on Interactive 3D Graphics, S. 123-126.
- Kim, Y.S. [1992], *Recognition of Form Features using Convex Decomposition*. In: Computer Aided Design 24 (9), S. 461-476.
- Klein, R. und Krämer, J. [1997], *Multiresolution Representations for Surface Meshes*. In: Proceedings of Spring Conference on Computer Graphics 1997, S: 57-66.
- Klinec, D. und Fritsch, D. [2003], *Towards Pedestrian Navigation and Orientation*. In: Proceedings of the 7th South East Asian Survey Congress, SEASC'03, Hong Kong.
- Kolbe, T. H., Gröger, G., Plümer, L. [2005], *CityGML – Interoperable Access to 3D City Models*. In: Proceedings of the First International Symposium on Geo-Information for Disaster Management GI4DM, Delft, Niederlande.
- Kolbe, T. [2006], *CityGML – Background and Design*. Präsentation von CityGML bei OGC OWS-4 in Reston, Virginia, 14 Juni 2006. (http://www.citygml.org/docs/Kolbe_CityGML_OWS-4_Kickoff_2006-06-14.pdf).

- Krus, M., Bourdot, P., Guisnel, F., Thibault, G. [1997], *Levels of Detail and Polygonal Simplification*, ACM's Crossroads, 3.4. (<http://www.acm.org/crossroads/xrds3-4/levdet.html>).
- Kruskal, J.B., Jr. [1956], *On the Shortest Spanning Subtree of a Graph and the Travelling Salesman Problem*. In: Proceedings of the American Mathematical Society, 7 (1956), S. 48-50.
- Kutulakos, K.N., Seitz, S.M. [2000], *A Theory of Shape by Space Carving*. In: International Journal of Computer Vision, Vol. 38 (3), S. 199-218.
- Lal, J. [2005], *Recognition of 3D Settlement Structure for Generalization*, Dissertation am Institut für Photogrammetrie und Kartographie, TU München.
- Lamy, S., Ruas, A., Demazeau, Y., Jackson, M., Mackaness, W., Weibel, R. [1999], *The Application of Agents in Automated Map Generalization*. In: Proceedings of the 19th Cartographic Conference, Ottawa.
- Lee, S.C., Nevatia, R. [2003], *Interactive 3D Building Modelling using a Hierarchical Representation*. In: IEEE Workshop on Higher-Level Knowledge in 3D Modelling and Motion Analysis (HLK '03), with ICCV '03, Nizza, Frankreich, S. 58-65.
- Li, Z. [1996], *Transformation of Spatial Representation in Scale Dimension*. In: International Archives of Photogrammetry and Remote Sensing, Vol. 31, Part B3/III, S. 453-458.
- Li, G. und Watson, B. [2001], *Semiautomatic Simplification*. In: ACM Symposium on Interactive 3D Graphics 2001, S. 43-48.
- Lindeberg, T. [1994], *Scale-Space Theory in Computer Vision*, Kluwer Academic Publishers, Boston, USA.
- Lindstrom, P. [2000], *Out-of-Core Simplification of Large Polygonal Models*. In: Proceedings of SIGGRAPH 2000, Vol. 34, ACM Press, New York, S. 259-262.
- Lindstrom, P. und Turk, G. [1998], *Fast and Memory Efficient Polygonal Simplification*. In: Proceedings of IEEE Visualization '98, S. 279-286.
- Lindstrom, P. und Turk, G. [1999], *Evaluation of Memoryless Simplification*. In: IEEE Transactions on Visualization and Computer Graphics, Vol. 5 (2), S. 98-115.
- Lindstrom, P. und Turk, G. [2000], *Image-Driven Simplification*. In: ACM Transactions on Graphics, Vol. 19 (3), S. 204-241.
- Lorenson, W. und Cline, H. [1987], *Marching Cubes: A High Resolution 3D Surface Construction Algorithm*. In: Proceedings of ACM SIGGRAPH 1987, S. 163-169.
- Luebke, D., Reddy, M., Cohen, J.D. [2002], *Level of Detail for 3D Graphics*, Morgan Kaufmann, USA.
- Low, K.-L. und Tan, T.-S. [1997], *Model Simplification using Vertex Clustering*, In: Proceedings of the 1997 Symposium on Interactive 3D Graphics, ACM SIGGRAPH 1997, ACM Press, New York, S. 75-82.
- Lu, Y., Gadh, R., Tautges, T.J. [1999], *Feature Decomposition for Hexahedral Meshing*. In: Proceedings ASME Design Automation Conference.
- Ma, R. [2004], *Building Model Reconstruction from LIDAR Data and Aerial Photographs*. Dissertation, The Ohio State University, USA.
- Mäntylä, M. [1988], *An Introduction to Solid Modeling*, Principles of Computer Science Series. Computer Science Press, Maryland, USA.
- Manandhar, D. und Shibasaki, R. [2002], *Auto-Extraction of Urban Features from Vehicle-Borne Laser Data*. In: Proceedings of the Joint International Symposium on Geospatial Theory, Processing and Applications, Ottawa, Kanada.
- Marefat, M. und Kashyap, R.L. [1990], *Geometric Reasoning for Recognition of Three-Dimensional Object Features*. In: IEEE Transactions on Pattern Analysis and Machine Intelligence 12 (10), S. 949-965.
- Marinov, M. und Kobbelt, L. [2005], *A Robust Two-Step Procedure for Quad-Dominant Remeshing*. In: Proceedings of Eurographics 2005.

- Mayer, H. [1998], *Model-Generalization of Building Outlines on Scale-Spaces and Scale-Space Events*. In: International Archives of Photogrammetry and Remote Sensing. Band 32 3/1. S. 520-536.
- Mayer, H. [2000], *Scale-Space Events for the Generalization of 3D-Building Data*. In: International Archives of Photogrammetry and Remote Sensing, Vol. 33, Part B 4/2., Amsterdam, Niederlande, S. 639-646.
- Mayer, H. und Reznik, S. [2006], *MCMC Linked with Implicit Shape Models and Plane Sweeping for 3D Building Facade Interpretation in Image Sequences*. In: Förstner, W. und Steffen, R. (Editors), ISPRS Commission III Symposium "Photogrammetric Computer Vision PCV06", IAPRS Vol. XXXVI, Part 3.
- Meng, L. [1997], *Automatic Generalization of Geographic Data*, Technical Report, <http://129.187.175.5/publications/meng/paper/generalization1997.pdf>.
- Meyer, U. [1989], *Generalisierung der Siedlungsdarstellung in digitalen Situationsmodellen*. Wissenschaftliche Arbeiten der Fachrichtung Vermessungswesen der Universität Hannover (159).
- Michaelsen, E., Soergel, U., Thoennessen, U. [2005], *Potential of Building Extraction from Multi-Aspect High-Resolution Amplitude SAR Data*. In: Joint Workshop of ISPRS / DAGM Object Extraction for 3D City Models, Road Databases and Traffic Monitoring CMRT, IAPRS, Vol. 36, Part 3/W24, S. 149-154.
- Mitra, N.J., Guibas, L.J., Pauly, M. [2006], *Partial and Approximate Symmetry Detection for 3D Geometry*. In: ACM Transactions on Graphics, Vol. 25, No. 3, S. 560-568.
- Müller, P., Vereenooghe, T., Wonka, P., Paap, I., Van Gool, L. [2006a], *Procedural 3D Reconstruction of Puuc Buildings in Xkipché*. In: Ioannidis, M., Arnold, D., Niccolucci, F., Mania, K. (Editors), Proceedings of the 7th Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST 2006)
- Müller, P., Wonka, P., Hägler, S., Ulmer, A., Van Gool, L. [2006b], *Procedural Modeling of Buildings*. In: ACM Transactions on Graphics, Vol. 25, No. 3, S. 614-623.
- Naylor, B.F. [1990], *Binary Space Partitioning Trees as an Alternative Representation of Polytopes*, In: CAD, Vol. 22, No. 4, S. 250-253.
- Neudeck, S. [2001], *Zur Gestaltung topografischer Karten für die Bildschirmvisualisierung*. Schriftenreihe des Studiengangs Geodäsie und Geoinformation der Universität der Bundeswehr München, Heft 74 (Dissertation).
- OpenGIS Consortium [2004], *Geographic Information – Geography Markup Language (GML)*. <http://www.opengis.net/gml/>
- Ortin, D. und Remondino, F. [2005], *Occusion-Free Image Generation for Realistic Texture Mapping*, In: International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 36, Part 5 / W 17, Mestre-Venedig, Italien (Auf CD-ROM).
- O'Rourke, J. [1998], *Computational Geometry in C (2nd Ed.)*, Cambridge University Press.
- Owen, J. D., Luebke, D., Govindaraju, N., Harris, M., Krüger, J., Lefohn, A. E., Purcell, T. J. [2005], *A Survey of General-Purpose Computation on Graphics Hardware*, In: Eurographics 2005, State of the Art Reports, August 2005, S. 21-51.
- Pauly, M., Niloy, M., Giesen, J., Gross, M., Guibas, L.J. [2005], *Example-Based 3D Scan Completion*. In: Proceedings of the Eurographics Symposium on Geometry Processing, Wien, Österreich, S. 23-32.
- Penard, L., Papanoditis, N., Pierro-Deseilligny, M. [2005], *3D Building Façade Reconstruction under Mesh Form from Multiple Wide Angle Views*. In: Proceedings of the ISPRS Working Group V/4 Workshop, 3D-Arch 2005 "Virtual Reconstruction and Visualizazion of Complex Architectures", Mestre-Venice, Italien.
- Pojar, E. und Schmalstieg, D. [2003], *User-Controlled Creation of Multiresolution Meshes*. In: Proceedings of ACM Symposium on Interactive 3D Graphics, S- 127.130.

- Pollefeys, M., Koch, R., Vergauwen, M., Van Gool, L. [2000], *Automated Reconstruction of 3D Scenes from Sequences of Images*. In: ISPRS Journal of Photogrammetry and Remote Sensing, Vol. 55, No. 4, S. 251-267.
- Powitz, B.-M. [1993], *Zur Automatisierung der Kartographischen Generalisierung topographischer Daten in Geo-Informationssystemen*. Wissenschaftliche Arbeiten der Fachrichtung Vermessungswesen der Universität Hannover, Nr. 185.
- Pratt, M. J., Wilson, P.R. [1985], *Requirements for Support of Form Features in a Solid Modeling System*, In: Technical Report R-85-ARPP-01, CAM-I Inc, Arlington, Texas.
- Prim, R.C. [1957], *Shortest Connection Networks and Some Generalizations*, In: Bell System Technical Journal, 36 (1957), S. 1389-1401.
- Puppo, E. und Scopigno, R. [1997], *Simplification, LOD and Multiresolution – Principles and Applications*. In: Eurographics '97 Tutorial Notes, Budapest, Ungarn.
- Rainsford, D. und Mackaness, W.A. [2002], *Template Matching in Support of Generalisation of Rural Buildings*. In: D. Richardson und P.v. Oosterom (Editors), *Advances in Spatial Data Handling 10th International, Symposium on Spatial Data Handling*. Springer-Verlag, Berlin, S. 137-152.
- Rau, J.Y. und Chen, L.C. [2003], *Robust Reconstruction of Building Models from Three-Dimensional Line Segments*. In: *Photogrammetric Engineering and Remote Sensing*, Vol. 69, S. 181-188.
- Rau, J.Y., Chen, L.C., Tsai, F., Hsiao, K.H., Hsu, W.C. [2006], *Automatic Generation of Pseudo Continuous LoDs for 3D Polyhedral Building Model*. In: *Innovations in 3D Geo Information Systems*, Springer-Verlag, Berlin.
- Regnault, N., Edwardes, A., Barrault, M. [1999], *Strategies in Building Generalization: Modelling the Sequence, Constraining the Choice*. In: *Progress in Automated Map Generalization – ACI '99*.
- Ribelles, J., Heckbert, P., Garland, M., Stahovich, T., Srivastava, V. [2001], *Finding and Removing Features from Polyhedra*. In: *American Association of Mechanical Engineers (ASME) Design Automation Conference*, Pittsburgh PA.
- Ripperda, N. und Brenner, C. [2006], *Reconstruction of Facade Structures using a Formal Grammar and rjMCMC*. In: Katrin Franke, Klaus-Robert Müller, Bertram Nickolay, Ralf Schäfer (Editors), *Pattern Recognition, Proceedings of the 28th Annual Symposium of the German Association for Pattern Recognition*, Berlin, S. 750-759.
- Ronfard, R., Rossignac, J. [1996], *Full-Range Approximation of Triangulated Polyhedra*. In: *Computer Graphics Forum, Proceedings of Eurographics 1996*, 15 (3), Poitiers, Frankreich.
- Rossignac, J., Borrel, P [1993], *Multi-Resolution 3D Approximations for Rendering Complex Scenes*. In: B. Falcidieno und T. Kunii (Editors), *Modeling in Computer Graphics: Methods and Applications*, Springer-Verlag, Berlin, S. 455-465.
- Rost, R. J. [2006], *OpenGL Shading Language (Second Edition)*, Addison Wesley.
- Rottensteiner, F. [2001], *Semi-Automatic Extraction of Buildings Based on Hybrid Adjustment using 3D Surface Models and Management of Building Data in a TIS*, Dissertation, Institut für Photogrammetrie und Fernerkundung, TU Wien.
- Samet, H. [2006], *Foundations of Multidimensional and Metric Data Structures*, Morgan Kaufmann Publishers.
- Sander, P., Snyder, J., Gortler, S., Hoppe, H. [2001], *Texture Mapping Progressive Meshes*. In: *Proceedings of ACM SIGGRAPH 2001*, Los Angeles, USA, S. 409-416.
- Schaefer, S. und Warren, J. [2003], *Adaptive Vertex Clustering using Octrees*. In: *Proceedings of SIAM Geometric Design and Computing 2003*, S. 491-500.
- Schilling, A., Blechschmied, H., Jasnoch, U. [2004], *Datenbank-basierte Visualisierung von Hamburg*. In: Coors, V., Zipf, A. (Editors), *3D-Geoinformationssysteme*, Wichmann Verlag.

- Schroeder, W.J., Zarge, A., Lorensen, W.E. [1992], *Decimation of Triangle Meshes*. In: Proceedings of SIGGRAPH 1992, S. 65-70.
- Schroeder, W.J. [1997], *Topology-Modifying Progressive Decimation Algorithm*. In: Proceedings of IEEE Visualization '97, S. 205-212.
- Segal, M. und Akeley, K. [2006], *The OpenGL Graphics System: A Specification (Version 2.1)*. <http://www.opengl.org/documentation/specs/version2.1/glspec21.pdf>
- Seitz, S.M., Dyer, C.R. [1999], *Photorealistic Scene Reconstruction by Voxel Coloring*. In: International Journal of Computer Vision, Vol. 35 (2), S. 151-173.
- Sester, M. und Klein, A. [1999], *Rule Based Generalization of Buildings for 3D-Visualization*. In: Proceedings of the 19th International Cartographic Conference of the ICA, Ottawa, Kanada, S. 214-224.
- Sester, M. [2000], *Maßstabsabhängige Darstellung in digitalen räumlichen Datenbeständen*. Habilitationsschrift, Deutsche Geodätische Kommission, Reihe C, Heft 544.
- Sester, M. und Brenner, C. [2004], *Continuous Generalization for Fast and Smooth Visualization on Small Displays*. In: Proceedings of the XXth Congress of the ISPRS, Vol. 35, Part B, Istanbul, Türkei.
- Shreiner, D., Woo, M., Neider, J., Davis, T. [2005], *OpenGL Programming Guide: The Official Guide to Learning OpenGL, Version 2*. Addison-Wesley Professional.
- Sörgel, U. [2003], *Iterative Verfahren zur Detektion und Rekonstruktion von Gebäuden in SAR- und InSAR-Daten*. Dissertation, Universität Hannover, Fakultät für Elektrotechnik und Informationstechnik.
- Sohn, G. und Dowman, I. J. [2001], *Extraction of Buildings from High-Resolution Satellite Data*, In: Automatic Extraction of Man-Made Objects from Aerial and Space Images (III), Swets & Zeitlinger B.V., Lisse, Niederlande, S. 345-355.
- Stadt, O. G., Gross, M. H. [1998], *Progressive Tetrahedralizations*. In: IEEE Visualization 1998, S. 397-402.
- Staufenbiel, W. [1973], *Zur Automation der Generalisierung topographischer Karten mit besonderer Berücksichtigung großmaßstäbiger Gebäudedarstellungen*, Wissenschaftliche Arbeiten der Fachrichtung Vermessungswesen der Universität Hannover.
- Steinle, E. [2005], *Gebäudemodellierung und -änderungserkennung aus multitemporalen Laserscanningdaten*. Dissertation. Deutsche Geodätische Kommission, Reihe C, Nr. 594.
- Stilla, U., Michaelsen, E., Jurkiewicz, K. [1998], *Structural Analysis of Right-Angled Building Contours*. In: International Archives of Photogrammetry and Remote Sensing, Vol. 32 3/1, S. 379-386.
- Stilla, U., Sörgel, U., Thönnessen, U. [2002], *Geometric Constraints for Building Reconstruction from InSAR data of Urban Areas*, In: Proceedings of the Joint International Symposium on Geospatial Theory, Processing and Applications, Ottawa, Kanada, S. 225-230.
- Strothotte, T., Preim, B., Raab, A., Schumann, J., Forsey, D. [1994], *How to Render Frames and Influence People*. In: Computer Graphics Forum (13) 3, Proceedings of EuroGraphics 1994, S. 455-466.
- Stüber, R. [2006], *Generalisierung von Gebäudemodellen unter Wahrung der visuellen Richtigkeit*, Heft 32 der Schriftenreihe für Kartographie und Geoinformation der Rheinischen Friedrich-Wilhelms-Universität Bonn.
- Su, B., Li, Z., Lodwick, G., Müller, J.-C. [1997], *Algebraic Models for the Aggregation of Area Features Based upon Morphological Operators*, In: International Journal of Geographical Information Science 11(3), S. 233-246.
- Suh, Y. S. und Wozny, M. J. [1997], *Interactive Feature Extraction for the Form Feature Conversion System*. In: Proceedings of the 4th ACM Symposium on Solid Modeling and Applications, Atlanta, GA USA, S. 111-122.
- Suveg, I. und Vosselman, G. [2004], *Reconstruction of 3D Building Models from Aerial Images and Maps*, In: ISPRS Journal of Photogrammetry & Remote Sensing 58, S. 202-224.

- Thibault, W.C. und Naylor, B.F. [1987], *Set Operations on Polyhedra Using Binary Space Partitioning Trees*, In: Proceedings of SIGGRAPH 1997, Anaheim, CA, USA. S. 153-162.
- Thiemann, F. [2001], *Parametrisierung, Modellierung und Ausgleichung von 3D Gebäudedaten für das Facility-Management*. Diplomarbeit, Institut für Geodäsie und Geoinformation, Technische Universität Berlin.
- Thiemann, F. [2002], *Generalization of 3D Building Data*. In: Proceedings of the Joint International Symposium on Geospatial Theory, Processing and Applications, Ottawa, Kanada.
- Thiemann, F. [2003], *3D-Gebäude-Generalisierung*, In: Kartographische Schriften, Band 7, Visualisierung und Erschließung von Geodaten, Seminar GeoVIS 2003, 27.-28. Februar 2003, Hannover, S. 185 - 192.
- Thiemann, F. und Sester, M. [2004], *Segmentation of Buildings for 3D-Generalisation*. In: Working Paper of the ICA Workshop on Generalisation and Multiple Representation, Leicester, UK.
- Thiemann, F. und Sester, M. [2005], *Interpretation of Building Parts from Boundary Representation*. In: Workshop on Next Generation 3D City Models (Foliensatz), Bonn.
- Thiemann, F. und Sester, M. [2006], *3D-Symbolization using Adaptive Templates*. In: Proceedings of the GICON 2006, Wien.
- Trotts, I. J., Hamann, B., Joy, K. I. [1999], *Simplification of Tetrahedral Meshes with Error Bounds*. In: IEEE Transactions on Visualization and Computer Graphics, 5(3), S. 224-237.
- Urena Camara, M.A., Ariza Lopez, F.J. [2005], *Generalization Process for Urban City-Block Maps*. In: Proceedings of the XXII International Cartographic Conference. La Coruna, Spain.
- Van Bakergem, W.D. und Obata, G. [1991], *Free Hand Plotting: Is it Life or is it Digital?*, In: CAD-Futures '91, G.N. Schmitt (Editor), Vieweg, S. 567-582.
- Van Kreveld, M. [2001], *Smooth Generalization for Continuous Zooming*. In: Proceeding of the ICA, Fourth Workshop on Progress in Automated Map Generalization, Peking, China.
- Vatti, B. [1992], *A Generic Solution to Polygon Clipping*, In: Communication of the ACM, July 1992, Vol. 35, No. 7. S. 57-63.
- Vorsatz, J., Rössl, C., Kobbelt, L., Seidel, H.-P. [2001], *Feature Preserving Remeshing*. In: Proceedings of Eurographics 2001, Manchester, UK.
- Wang, X., Totaro, S., Taillandier, F., Hanson, A., Teller, S. [2002], *Recovering Facade Texture and Microstructure from Real-World Images*. In: International Archives of the Photogrammetry. Remote Sensing and Spatial Information Sciences, Vol. 34, 3A, S. 381-386.
- Watt, A. [2000], *3D Computer Graphics*, 3. Auflage, Addison-Wesley, Amsterdam.
- Weibel, R. [1996], *A Typology of Constraints to Line Simplification*. In: Proceedings of the 7th International Conference on Spatial Data Handling, S. 533-546.
- Weiler, K. [1985], *Edge-Based Data Structures for Solid Modeling*. In: Curved-Surface Environments, IEEE Computer Graphics & Applications, Vol. 5, No. 1, S. 21-40.
- Werner, T. und Zisserman, A. [2002], *New Techniques for Automated Architectural Reconstruction from Photographs*. In: Proceedings of the 7th European Conference on Computer Vision, Vol. 2, S. 541-555.
- Wolf, H. [1975], *Ausgleichsrechnung – Formeln zur praktischen Anwendung*, Dümmler Verlag, Bonn.
- Wolf, M. [1999], *Photogrammetric Data Capture and Calculation for 3D City Models*. In: Fritsch, Spiller (Editors), Photogrammetric Week '99, Wichmann Verlag, Heidelberg, S. 305-312.
- Woo, T. [1982], *Feature Extraction by Volume Decomposition*. In: Proceedings of the CAD/CAM Technology in Mechanical Engineering Conference.
- Wu, M.C., Liu, C.R. [1996], *Analysis on Machine Feature Recognition Techniques based on B-Rep*. In: Computer Aided Design, Vol. 28 (8), S. 603-616.

Lebenslauf

Persönliche Angaben

Martin Kada

Geboren am 4. November 1972 in Rendsburg

Schulausbildung

1979 – 1983 Kelterschule, Neckarrems

1983 – 1990 Lise-Meitner-Gymnasium, Remseck

1990 – 1991 Central High School, Grand Forks, Nord-Dakota, USA

1991 – 1994 Lise-Meitner-Gymnasium, Remseck

Studium

1994 – 2000 Diplomstudium Informatik an der Universität Stuttgart

Beruf

2001 – 2007 Wissenschaftlicher Mitarbeiter am Institut für Photogrammetrie der Universität Stuttgart