

# Automatische Extraktion von Fenstern in 3D Punktwolken mittels einer hierarchischen Methode

SVEN SCHNEIDER<sup>1</sup> & VOLKER COORS<sup>1</sup>

*Zusammenfassung: Energetische Simulationen im urbanen und besonders im innerstädtischen Raum erfahren eine immer größere Nachfrage, z.B. um nachhaltige Stadtentwicklung zu betreiben. 3D Stadtmodelle sind hierfür gut geeignet. Diese können durch Simulationssoftware wie SimStadt genutzt werden, um Wärmebedarfsanalysen zu simulieren. Viele amtliche Daten sind jedoch nicht mit den nötigen Kennwerten ausgestattet. Zu diesen gehören u.a. die Stockwerksanzahl und das Fenster-Fassaden-Verhältnis von Gebäuden. Mit der hier vorgestellten Methode lassen sich diese Kennwerte über die detektierten Fenster ableiten. Zum Detektieren der Fenster werden zunächst Fassaden (Ebenen) in Punktwolken gesucht. Danach folgen das Invertieren der Fassadenpunktwolke und die Konturverfolgung der Fenster. Zuletzt werden Konturen gelabelt und die Flächen dieser anhand ihrer kleinsten umschreibenden Rechtecke berechnet. Ergebnisse für die Fensterdetektion liegen, je nach Fassade, bei einer Treffgenauigkeit von 80-90%. Mit Hilfe der Position und der Größe der Fenster lassen sich somit die entsprechenden Kennwerte für Simulationen ableiten.*

## 1 Einleitung

Rund 40% des Energiebedarfs einer Stadt entsteht durch den Gebäudesektor. In privaten Haushalten, sind ca. 85% der Energie für den Heizbedarf notwendig (BMWI 2014). Durch die Energiewende wird es immer wichtiger, dass Neubauten energieeffizient gebaut werden und Eigentümer und Vermieter Empfehlungen zur Sanierung ihrer Gebäude erhalten, um den CO<sub>2</sub> Ausstoß nachhaltig zu reduzieren (BMWI 2011). In diesem Zuge sind Simulationswerkzeuge wie z.B. SimStadt (NOUVEL et al. 2015) geeignete Werkzeuge um den Energie- und Wärmebedarf solcher baulicher Maßnahmen zu simulieren.

Amtliche Daten sind oft nicht mit allen relevanten Sachdaten und Kenngrößen versehen, die für solche Simulationen notwendig sind (BILJECKI et al. 2015; NOUVEL et al. 2017). In vielen Fällen besteht daher die Notwendigkeit zur Anreicherung von Datenbeständen mit Sachdaten aus anderen Quellen. Wichtige Sachdaten zur Simulation, wie Stockwerksanzahl oder dem Fenster-Fassaden-Verhältnis, sind oft nicht oder nur teilweise in Datenbanken erfasst. Durch Annahme eines quartierübergreifenden Mittelwerts (z.B. für das Fenster-Fassaden-Verhältnis) werden in Simulationen akzeptable Ergebnisse erreicht, jedoch ist die Aussagekraft für kleine Einheiten (z.B. einzelne Gebäude), mit relativ großen Unsicherheiten behaftet.

Für eine möglichst realistische energetische Gebäudebetrachtung ist das Verhältnis von Fensterfläche zur Gesamtgebäudefläche relevant, da Fenster und Fassade in der Regel erheblich unterschiedliche Wärmedurchgangskoeffizienten (U-Werte) aufweisen (MONIEN et al. 2016). Die Kenntnis der tatsächlichen Fensterflächenanteile einer Fassade liefern daher genauere Ergebnisse bei einer Wärmebedarfsanalyse (MONIEN et al. 2017).

---

<sup>1</sup> Hochschule für Technik Stuttgart, Zentrum für Geodäsie und Geoinformatik, Schellingstr. 24, D-70174 Stuttgart, E-Mail: [sven.schneider, volker.coors]@hft-stuttgart.de

Durch automatisierte Auswerteverfahren von photogrammetrisch gewonnenen Daten, beispielsweise Schrägluftbilder aber auch 3D Punktwolken, können relevante Sachdaten ergänzt werden. Daher ist es möglich, Stadtmodelle in einem niedrigen LOD (1 oder 2) zu nutzen obwohl aus diesen geometrischen Daten (i.d.R. im CityGML vorliegend) die nötigen Kennwerte i.d.R. nicht abgeleitet werden können.

Zu dem oben beschriebenen Zweck wird in diesem Beitrag eine hierarchische Methode zur Extraktion von Fenstern an Gebäudefassaden präsentiert. Gebäudemodelle können mit dem vorgestellten Verfahren mit relevanten Sachdaten angereichert werden und für Simulationen genutzt werden.

Der Artikel gliedert sich folgendermaßen: Sektion 2 beschreibt die Problemstellung dieser Arbeit und gibt Kontext zu bestehenden Methoden; Sektion 3 beschreibt die zu Grunde liegenden Daten; Sektion 4 beschreibt die angewendete Methode und zeigt erste Zwischenergebnisse für jeden Verarbeitungsschritt; Sektion 5 diskutiert und fasst die Arbeit zusammen.

## 2 Problemstellung

Da Stadtmodelle mit detaillierten Informationen über Fenster und Stockwerksanzahl oft nur teilweise oder auch gar nicht vorliegen, besteht eine große Notwendigkeit (z.B. für Simulationen) diese Informationen auf anderen Wegen zu ergänzen. Neben Schrägluftbildern stellen auch sogenannte Mobile Mapping Verfahren mögliche Alternativen zur Verfügung. Mobile Mapping Daten werden in der Regel durch Fahrten mit einem Fahrzeug und entsprechenden Laserscannern durchgeführt (siehe Sektion 3).

Die Herausforderung, Fenster in Fassaden zu detektieren, ist aus diversen Gründen schwierig:

- (i) Oft sind Fenster gar nicht oder nur teilweise in Punktwolken erfasst, da der Laserstrahl die Fensterscheibe durchdringt oder durch spiegelnde Reflexion vom Detektor weg reflektiert wird.
- (ii) Fenster sind i.d.R. von der Fassade leicht nach hinten versetzt (Abbildung 1). Dieser Abstand variiert in den betrachteten Punktwolken je nach Gebäudetyp und Baujahr von ca. 15 cm, bis zu 35 cm.
- (iii) Ähnlich wie der Versatz der Fenster, variiert auch die Vielfalt der Fensterformen, daher funktioniert eine Detektion über Templates nur mit relativ hohem Aufwand.
- (iv) Rollläden stellen eine zusätzliche Herausforderung dar, da sie viele ungewissen Zustände haben können (ganz offen, geschlossen, halb offen und alles dazwischen).
- (v) Letztlich kommen zu diesen speziellen Herausforderungen noch die üblichen messtypischen Einflussfaktoren hinzu, z.B. Verdeckungen von Fassaden und Fensterteilen durch Vegetation, parkende Fahrzeuge und / oder Laternenmasten.

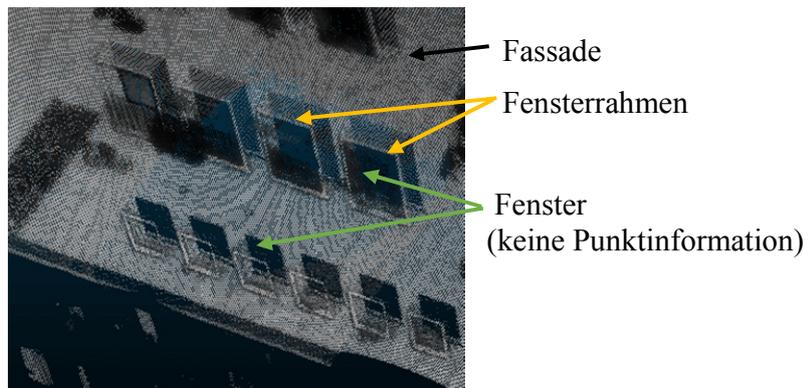


Abb. 1: Gebäudefassade betrachtet von der Innenseite. Die Fensterrahmen sind gut zu erkennen und der Versatz der Fenster zur Fassade. Hier liegen die Fenster vor der Fassade, da dieser Screenshot der Punktwolke von der Innenseite gemacht wurde

## 2.1 Lösungsansätze aus der Literatur

Einige der bekannten Ansätze zur Detektion von Fenster in Punktwolken nutzen bildgebende Verfahren ausschließlich oder unterstützend, d.h. dass Fassaden mit RGB-D Sensoren aufgenommen oder zusätzlich zu Laserscannern mit Kameras fotografiert wurden (z.B. ALI et al. 2007; NEUHAUSEN et al. 2018; TRINH et al. 2009; ZHANG & ZAKHOR 2014). Verfahren, die unabhängig von Kamerabildern sind und ausschließlich auf Laserscanndaten basieren sind in den letzten Jahren gestiegen. Die Methode von TUTTAS UND STILLA (2013) erzielt Resultate mit einer Klassifiziergenauigkeit von knapp 100% für Fassaden mit gleichmäßigen Strukturen. Für Häuserblocks in Wohngebieten, mit abwechselnden und kurzen Segmenten von Fenster, erzielt dieses Verfahren auf Grund der genutzten Fourier Transformation schlechte Resultate. Andere Methoden nutzen Verfahren des maschinellen Lernens um Fensterformen zu lernen und in Fassaden zu detektieren (NGUATEM et al. 2014). Dadurch ist es einerseits möglich Fenster mit unterschiedlichen Formen zu detektieren (z.B. mit Rundungen), andererseits verliert man auf diesem Wege Flexibilität, da nur gelernte Muster erkannt werden können. Das Erstellen von gelabelten Trainingsdaten ist hier ebenfalls problematisch und limitierend.

## 3 3D-Laserpunktwolken

### 3.1 Erfassung der Daten

Laserscan-Daten wurden im Rahmen des WeBest Projekts (HFT Stuttgart; MONIEN et al. 2016) von der Firma IGI im Auftrag von EUROSENSE akquiriert. Es wurden vier repräsentative Quartiere mit insgesamt etwa 500 Wohngebäuden in unterschiedlichen Baualtersklassen der Stadt Essen ausgewählt. Bei der Messfahrt wurde auf eine konstant niedrige Geschwindigkeit ( $< 30$  km/h) geachtet. Der eingesetzte Laser Z+F Profiler (Genauigkeit ca.  $< 1$  mm bei einer Distanz von 25 m) ermöglichte die Aufnahme eines  $360^\circ$  großen Field-of-Views mit einer Scanrate von bis zu 1 Million Punkten pro Sekunde. Gleichzeitig kam eine Novatel GPS-Antenne L1/L2 Model 512 und eine ApplanixIMU 200Hz zum Einsatz (Aufbau siehe Abbildung 2). Da der Sensor

relativ weit oben befestigt war, führte dieses Verfahren zu sehr hochauflösenden, georeferenzierten 3D Punktwolken mit relativ wenig Verdeckungen auf den Fassaden. Dennoch konnte nicht verhindert werden, dass auf einigen Fassaden Lücken durch geparkte Fahrzeuge, Masten von Straßenlaternen und Bäumen in den Punktwolken vorzufinden waren.

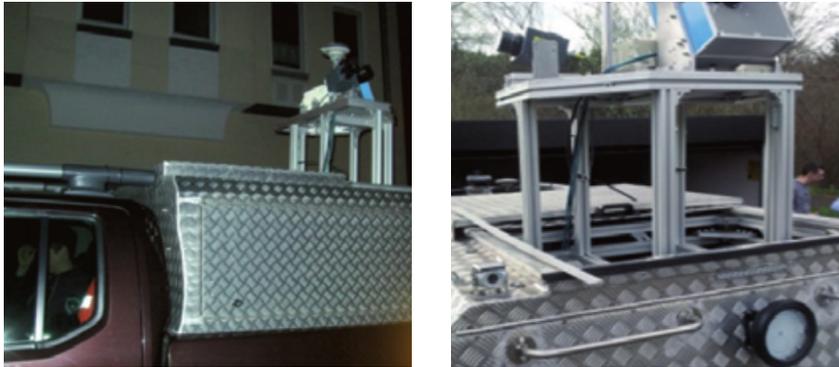


Abb. 2: Mobile Mapping Plattform mit GPS Antenne, Laserscanner, IMU und Thermographie- Kamera (in dieser Arbeit nicht verwendet). Quelle: MONIEN et al. (2016)

### 3.2 Merkmale der Punktwolken

Die Punktwolken sind hochauflösend und geben die Umgebung sehr detailgetreu wieder. Die Auflösung variiert mit der Distanz zum Laserscanner, d.h. die höchste Auflösung wird im Nahbereich des Lasers erreicht (z.B. der Straße, 2400-2800 Punkte pro qm) und die niedrigste Auflösung wird an den höchsten und weitest entfernten Punkten erreicht (z.B. das obere Ende der Fassade, 450-800 Punkte pro qm). Verglichen mit typischen Laserpunktwolken aus der Fernerkundung (z.B. durch amtliche Befliegungen), welche oft Werte von ca. 4 Punkten pro qm erreichen, sind die hier erreichten Auflösungen selbst am niedrigen Ende noch extrem hochauflösend. Abbildung 3 (oben) zeigt die Verteilung der Punktdichte in Abhängigkeit des Abstands und eine Fassade der Originalpunktwolke mit unterschiedlichen Intensitätswerten (unten). Punktwolken sind im LAS Format abgespeichert und die Koordinaten sind im DHDN GK Format vorhanden. Eine typische Punktwolkendatei hat zwischen 10 und 14 Millionen Punkte und eine Dateigröße von 400-500 Megabyte. Da diese große Anzahl an Punkten zu einer sehr langsamen Verarbeitung führen würde, wird die Punktwolke im Schritt der Datenvorverarbeitung zuerst in der Auflösung reduziert (Sektion 4).

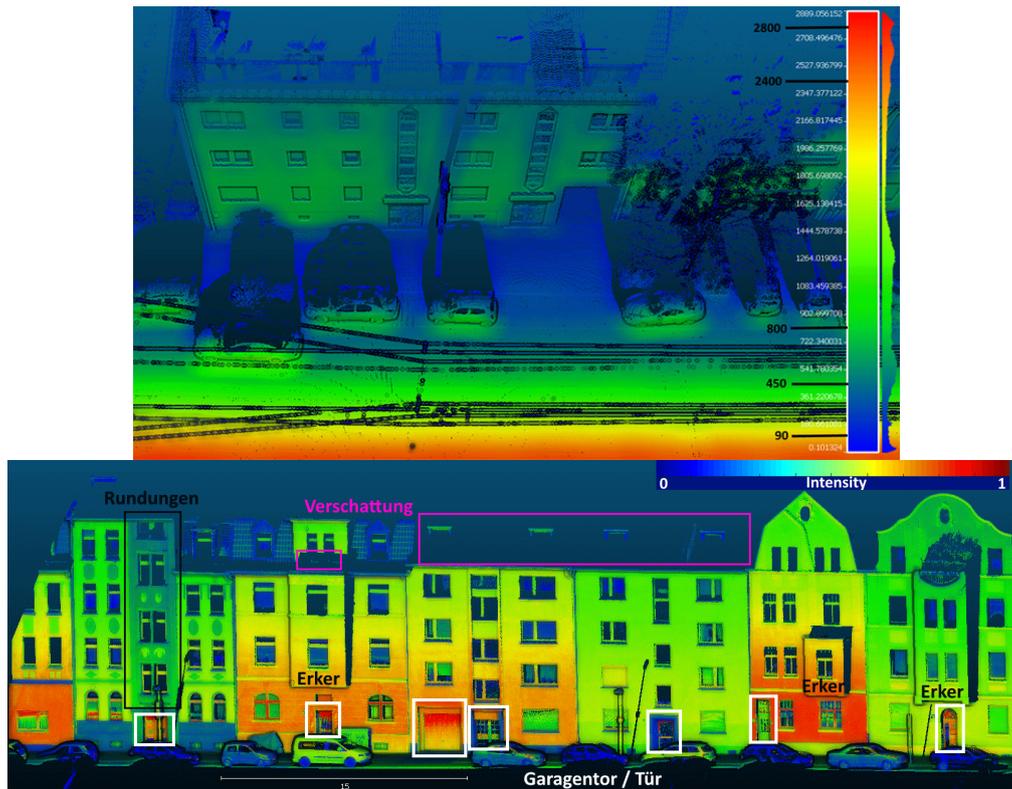


Abb. 3: Ausschnitt einer Punktwolke in Falschfarbendarstellung. Die Punktdichte pro qm ist im oberen Bild zu sehen. Intensitätswerte der reflektierten Strahlen sind unten zu sehen. Die Annotationen wie Erker, Garagentor, etc. sind für spätere Diskussionen relevant

## 4 Methode und Resultate

Die Methode ist in fünf große Schritte aufgeteilt welche in Abbildung 4 schematisch dargestellt sind. Die einzelnen Verarbeitungsschritte werden in den folgenden Sektionen detailliert beschrieben.

Die Methode wurde in C++ implementiert unter Verwendung gängiger Software Bibliotheken wie z.B. die Point Cloud Library (PCL; RUSU & COUSINS 2011), welche die Grundlage für viele der im Folgenden verwendeten Methoden darstellt.

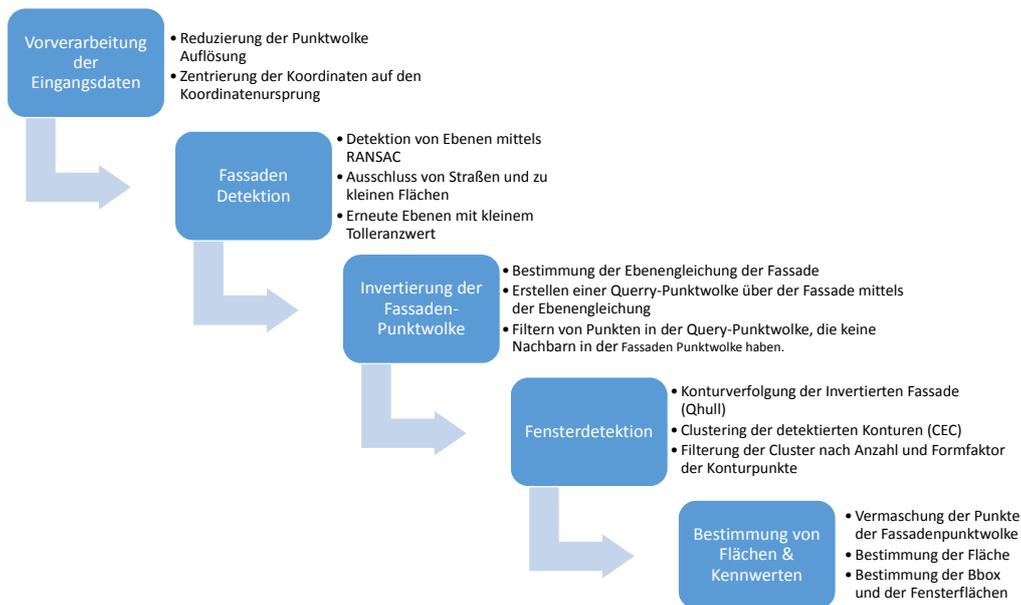


Abb. 4: Übersicht der Prozesskette zur Bestimmung der Fenster in Fassaden

## 4.1 Vorverarbeitung der Punktwolke

### 4.1.1 Zentrierung der Punktwolkenkoordinaten

Die Punktwolken im LAS Format sind im als Datentyp Double gespeichert, allerdings unterstützt die PCL nur den Float Datentyp. Würden die LAS Daten per Typumwandlung direkt in ein PCL Punktwolkenobjekt gespeichert werden, würden signifikante Stellen und Präzision verloren gehen. Dies würde zu Artefakten führen, welches für die weitere Verarbeitung problematisch wäre (Abbildung 5).

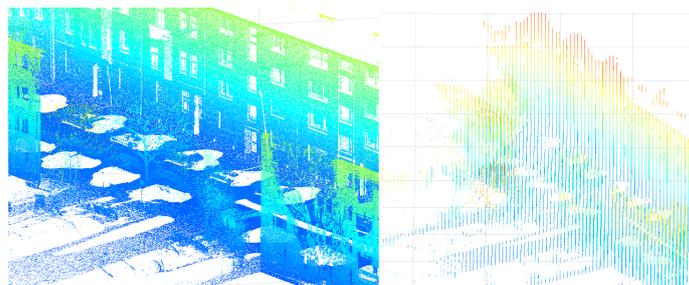


Abb. 5: Darstellung der Artefakte (nur rechts) der Punktwolke im Format Double (links, original) und Float (rechts, konvertierte Punktwolke) nach Durchführen einer einfachen Typkonvertierung.

Um diese Artefakte zu vermeiden, wird die Punktwolke zuerst auf den Koordinatenursprung zentriert, d.h. nach der Zentrierung hat die kleinste Koordinate in  $x,y,z$  (in Kartesischen Koordinaten), jeweils den Wert Null. Dadurch werden bis zu sieben Stellen vor dem Dezimalpunkt gesparrt, welche für die Nachkommastellen und entsprechend die Genauigkeit entscheidend sind.

#### 4.1.2 Reduzierung der Auflösung

In diesem Schritt wird die sehr hohe Auflösung der Punktwolke reduziert. Es ist von Vorteil sehr detaillierte Daten zu haben, jedoch ist die Datenmenge für eine zügige Bearbeitung zu groß. Die Punktauflösung wird durch ein Voxel-Grid Filter der PCL reduziert, in dem die Punktwolke in viel größere Voxel aufgeteilt, und der Mittelwerte jedes Voxel als gefilterter Punkt übernommen wird. Werte von 4-6 cm<sup>3</sup> per Voxel ergaben für die verwendeten Datensätze eine deutliche Reduzierung (von ca. 13 Mio. auf ca. 2 Mio. Punkte) bei sehr geringer Reduzierung der Qualität. Die reduzierte Punktwolke ist selbst nach der Punktreduzierung immer noch sehr hochauflösend und ausreichend für die weitere Verarbeitung.

#### 4.2 Detektion von Fassaden in der Punktwolke

Zur Detektion der Fassade wird angenommen, dass Fassaden näherungsweise flache Ebenen sind. Diese Annahme ist für den Großteil der Fassaden legitim und führt zu guten Ergebnissen. Ebenen in der Punktwolke werden mittels RANdom SAMple Consensus (RANSAC) Algorithmus in der Punktwolke detektiert. Ein Toleranzwert von 40 cm wird für die Detektion angenommen. D.h. alle Punkte, die innerhalb dieses Toleranzwertes sind, sind Punkte auf der gesuchten Ebene (Abbildung 6). Es werden jedoch auch andere Ebenen als die gewünschten Fassaden gefunden (z.B. Straßenebenen). Diese werden durch ermitteln des Normalvektors der Ebene (besonders nützlich für die Straßenebene), als auch über die Anzahl der Punkte in der Ebene entfernt. D.h. besteht eine Ebene aus weniger als 15 % der gesamten Punktwolke (hier ist die bereits reduzierten Punktwolke gemeint), wird die Ebene nicht als Fassade klassifiziert und von der weiteren Auswertung ausgeschlossen.



Abb. 6: Extrahierte Fassade durch Ebenen Detektion mittels RANSAC (Toleranz 40 cm). Viele Fenster, aber nicht alle, haben eine Punktinformation (meist rot). Die Farbskala entspricht der reflektierten Intensität des Laserpulses

Anschließend wird eine neue Ebenendetektion für die restlichen Punkte in der Punktwolke durchgeführt. Hier wird der Toleranzwert für die Ebenendetektion auf 15 cm reduziert. Dies führt dazu, dass nun nahezu alle Fenster ohne Punktinformation sind (Abbildung 7). In diesem Schritt sind die Fenster im Grunde bereits detektiert, dies wird durch Visualisierungen dieses Schrittes deutlich. Da die Fenster zum jetzigen Zeitpunkt allerdings durch Absenz von Information gekennzeichnet sind, werden weitere Verarbeitungsschritte notwendig.

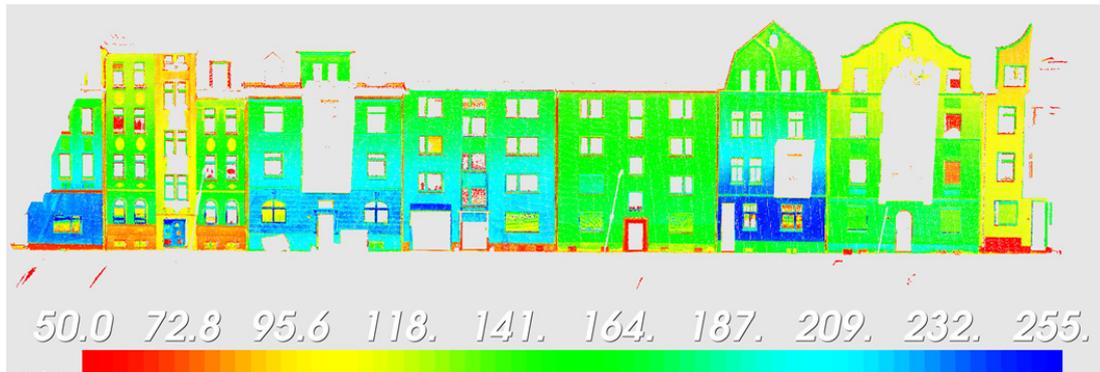


Abb. 7: Zweite Ebenen Detektion mit Toleranz von 15 cm. Fenster sind nun überwiegend ohne Punktinformation. Die Farbskala entspricht der reflektierten Intensität des Laserpulses

Es gibt jedoch Einschränkungen bei der Detektion der Fassadenebenen. Dies ist zum einen die eingangs beschriebene Annahme, dass Fassaden in einer Ebene liegen müssen, welche mit der Ebenengleichung ( $Ax + By + Cz + D = 0$ ) beschrieben werden kann. Eine weitere Einschränkung dieser Methode ist, dass hervorstehende Fassadenteile (z.B. Erker) nicht erkannt werden, da sie nicht innerhalb der Toleranzwerte des RANSAC Verfahrens für die Ebenendetektion liegen. Dies kann jedoch relativ leicht behoben werden, indem eine Ebenendetektion in einem Abstand  $\Delta d$  vor der Fassade durchgeführt wird (wobei  $\Delta d > \text{RANSAC Toleranzwert}$  ist).

### 4.3 Invertierung der Fassadenpunktvolke

Im vorherigen Schritt wurden die Fenster durch fehlende Informationen der Fenster quasi detektiert. Um konkrete Punktinformationen über die Fenster zu erhalten, wird die Fassadenpunktvolke nun invertiert und folgt damit konzeptionell dem Ansatz zum Invertieren eines Binärbildes. Zum Invertieren wird eine gleichmäßige Struktur aus Punkten über die Fassade gelegt (hier im Abstand von 25 cm), nachdem die Ebenengleichung der Fassade über ein RANSAC Verfahren ermittelt wurde. Die Güte der Fensterdetektion hängt von der Auflösung der Punkte in der Gitterpunktvolke ab: je hochauflösender das Gitter ist, desto genauer werden die Fenster abgebildet. Die Punktvolke wird in die Datenstruktur eines Kd-Baums überführt, welche für die folgende Nachbarschaftssuche notwendig ist. Für jeden Punkt in der Gitterpunktvolke, wird an der gleichen Position in der Fassadenpunktvolke (und dessen unmittelbaren Umgebung), eine Suche nach Fassadenpunkten gestartet. Wird ein Fassadenpunkt in der Fassadenpunktvolke in diesem Suchbereich gefunden, so wird dieser Punkt aus der Gitterpunktvolke entfernt; andernfalls wird der Punkt in der Gitterpunktvolke gespeichert bzw. nicht eliminiert (Abbildung 8). Dies führt dazu, dass hauptsächlich die Fenster übrigbleiben und Fassadenpunkte eliminiert werden. Ein wichtiger Parameter in diesem Verarbeitungsschritt ist der Suchradius für die Nachbarschaftssuche. Für die in dieser Studie getesteten Punktvolken wurde ein Radius mit  $r = 15 \text{ cm}$  genutzt.



Abb. 8: Invertierte Punktwolke durch die Gitterpunktwolke. Außerhalb der Fassade sind nun viele Punkte (oberhalb und unterhalb), die keine Fenster darstellen. Fenster haben nun Punktinformationen, die Fassade ist im Grunde eliminiert

#### 4.4 Fensterdetektion mittels Konturverfolgung

In diesem Schritt wird die invertierte Fassade als Eingangssignal für die Detektion der konkaven Hülle der Fenster und der Fassade genutzt, welche der Kontur eines Objektes (Fenster) entspricht (Abbildung 9). Algorithmen hierfür werden ebenfalls aus der PCL genutzt, welche auf den Quickhull Algorithmus (Qhull, BARBER et al. 1996) zurückgreift. Ein wichtiger Parameter für die Hüllendetektion mit Qhull ist  $\alpha$ . Dieser Parameter steuert wie detailliert die Kontur verfolgt wird. Kleine Werte (ca. 0.15), sorgen für relativ detaillierte Konturen, große Werte ( $> 0.5$ , mit einem Maximalwert von 1), sorgen für grobe, approximierte Konturen. Mit großen Parameterwerten bleiben mehr Fenster unentdeckt als bei kleinen Werten. Z.B. werden Fenster am Rande des Gebäudes mit zur Fassadenkontur gezählt, d.h. Details und kleinere Fenster gehen verloren. Ein zu kleiner Parameter sorgt dafür, dass zu viele kleine Details erkannt werden, z.B. vereinzelte Punkte (Rauschen, Äste von Bäumen und Laternenmaste). In der Implementierung werden jedoch im nächsten Verarbeitungsschritt kleine Objekte gefiltert. Daher ist eine leichte Übersegmentierung weniger problematisch als eine Untersegmentierung.

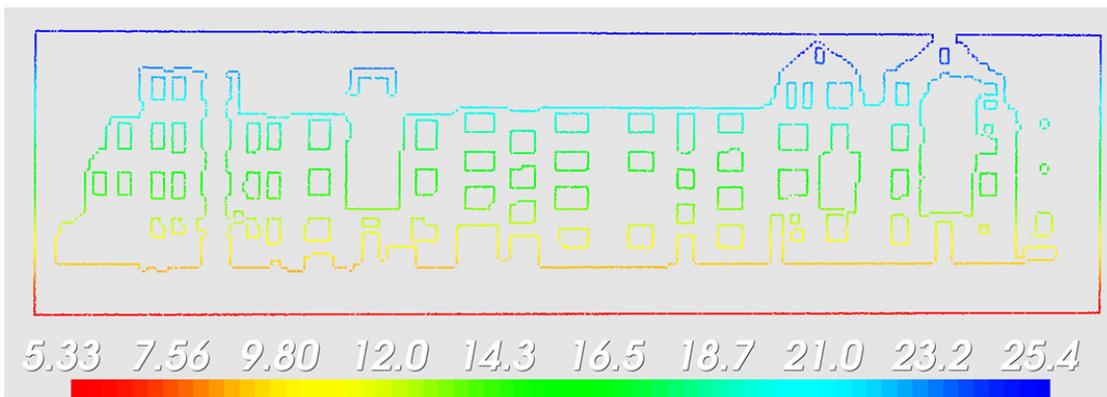


Abb. 9: Konturverfolgung der invertierten Gitterpunktwolke. Der äußere Rahmen ist in einem Nachverarbeitungsschritt leicht zu entfernen. Übrig bleiben die Umrisse der Fassade und der Fenster

#### 4.5 Gruppierung von Punkten (Clustering)

Als vierter Schritt, wird ein „Conditional Euclidean Clustering“ Algorithmus (CEC, RUSU 2009) auf die detektierten Umrisse der Fenster angewendet. Dabei werden Objekte, die keine Fenster darstellen, weitestgehend entfernt und jedem Objekt wird ein eindeutiges Label zugeordnet (Abbildung 10).

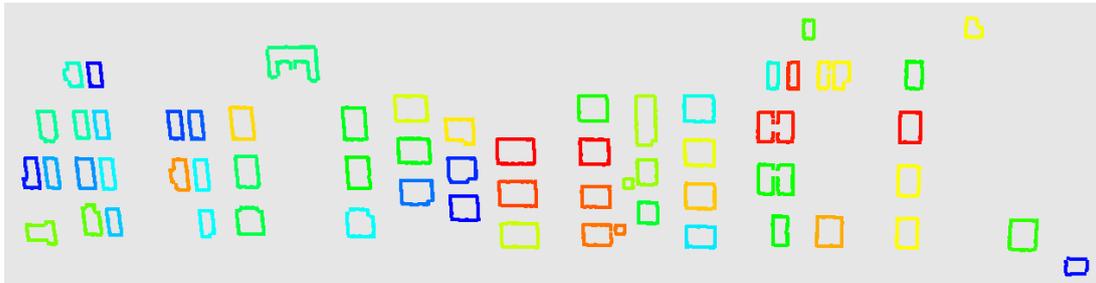


Abb. 10: Ergebnis des CEC Algorithmus mit zufällig eingefärbten Objekten / Fenstern.

Das Entfernen kleiner Objekte erfolgt über die Anzahl der Punkte per Cluster. Ein geeigneter Wert für diesen Parameter wurde empirisch ermittelt und liegt zwischen Werten von 35 und 180 Punkten, welche robuste Clustergrößen mit unterschiedlichen Punktwolken liefern. Sollten sich die Eingangsaufösungen der Punktwolken gravierend ändern, muss dieser Parameter ggf. angepasst werden. Weitere Parameter sind der Suchradius für den Normalenvektor suche ( $r = 7,5 \text{ cm}$ ) sowie „Cluster Toleranz“ (30 cm), also der maximale Abstand zwischen einem Punkt der bereits zu einem Cluster gehört und einem Kandidatenpunkt. Nur wenn der Abstand kleiner ist, werden die weiteren Konditionen überprüft und der Punkt entsprechend hinzugefügt oder abgewiesen. Ein abgewiesener Kandidatenpunkt kann aber zu einem späteren Zeitpunkt trotzdem zu diesem Cluster hinzugefügt werden, sollten alle Bedingungen zwischen dem Kandidatenpunkt und einem anderen Punkt des Clusters erfüllt sein. Die hier genutzten Bedingung zur Bewertung zum Aufnehmen eines Punktes waren (neben dem maximalen Abstand) die gleiche Orientierung der Flächennormalen.

#### 4.6 Bestimmen von Fensterflächen und Kennwerten

In diesem Schritt werden die Fensterflächen und Fassadenflächen berechnet. Bei den Fassadenflächen ist es allerdings einfacher und unter Umständen genauer diese Information aus den Gebäudemodelldaten (also dem CityGML) abzuleiten.

Die Fensterflächen werden approximiert, in dem das kleinste umschreibende Rechteck (BoundingBox) für jedes Cluster aus Fensterumrisspunkten ermittelt wird. Dazu wird ebenfalls der Qhull Algorithmus aus der PCL verwendet. Die vier Eckpunkte des Fensters werden durch diesen Algorithmus bestimmt und die Fläche wird entsprechend über die Differenzen der Breiten- und Höhenkoordinaten berechnet. Durch diese Methode werden Fensterflächen mit Rundungen jedoch überbewertet, was der Einfachheit des Verfahrens geschuldet ist. Es kann jedoch angenommen werden, dass dies für die meisten Fassaden mit Fenstermischformen und für die meisten Anwendungsfälle, diese Diskrepanz keinen großen Einfluss auf die Ergebnisse hat.

Die Eckpunkte des detektierten Rechtecks können im Folgenden dann für weitere Berechnungen genutzt werden oder bei Bedarf auch als Polygonzug gespeichert werden (z.B. als CityGML), als Ergänzung zu einem bestehenden Gebäudemodell.

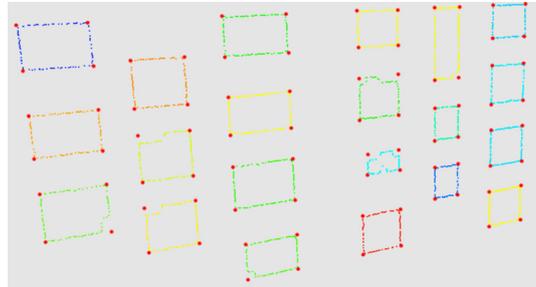


Abb. 11: Vergrößerter Teilausschnitt der Fenster mit Konturen mit überlagerten Eckpunkten (rote Punkte) der bestimmten BoundingBoxen. Unterschiedliche Farben der Fenster entsprechen unterschiedlichen Labels, die während des CEC bestimmt wurden

#### 4.7 Auswertung

Die oben beschriebene Fassade (Abbildung 3) wurde zur ersten Evaluierung der Methode genutzt. Da eine Limitierung der Methode darin besteht, Fenster in hervorstehenden Objekten wie Erkern nicht erkennen zu können, werden Fenster, die sich an solchen Stellen befinden nicht in der Auswertung berücksichtigt. Lediglich die Fenster, die nach der Ebenendetektion noch sichtbar sind, werden berücksichtigt (in Magenta eingefärbt, Abbildung 12). Fenster, die mit einem weißen Punkt markiert sind, wurden richtig erkannt (vgl. Abbildung 10). Die quantitative Auswertung zeigt recht hohe Werte für die Treffgenauigkeit und den F1-score ( $2 \cdot TP / (TP + FP + TN + FN)$ ), siehe Tab. 1).

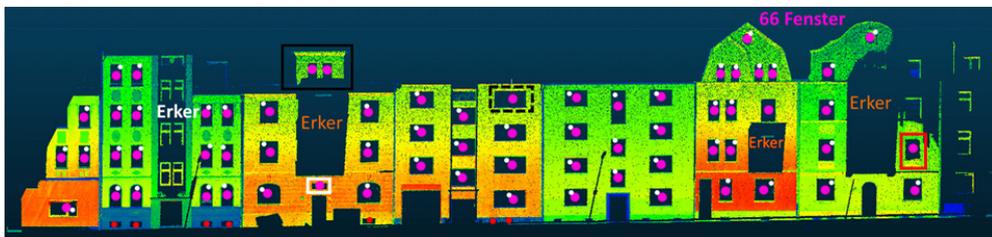


Abb. 12: Fassade mit markierten Fenstern (Magenta), welche für die Auswertung berücksichtigt werden. Rot markierte Fenster werden nicht erkannt, da sie unterhalb der minimalen Höhe zur Suche liegen

Tab. 1: Übersicht der Auswertung für die obere Fassade. Es wurde eine Treffgenauigkeit (Accuracy) von 87 % und ein F1-score von 92 % erreicht. Als „kein Fenster“ werden Türen bzw. Garagentore bezeichnet

		Tatsächlich	
		Fenster	kein Fenster
Prädiziert	Fenster	TP=60	FP=4
	kein Fenster	FN=6	TN=7

Die Auswertung zeigt auch, dass keine Türen oder Garagentore als Fenster erkannt wurden (7 TN). Dies wurde zum einen durch die Anzahl der erlaubten Punkte in einem Cluster beschränkt und zum anderen über die Höhe ausgeschlossen. Einige Fenster wurden jedoch nicht detektiert. So sind die Fenster im Verschattungsbereich (siehe schwarzes Rechteck in Abbildung 12) nicht detektiert worden, jedoch wurde der ganze obere Bereich der Fassade als Fenster detektiert was zu einer Fehldetektion führt (FP). Andere Fenster (schwarze, gestricheltes Rechteck und rotes Rechteck) wurden nicht erkannt, da durch Rauschen die Fensterkontur zur Fassadenkontur gezählt wurde und bei dem anderen Fenster die Fassade bereits lückenhaft detektiert wurde (womöglich durch Verschattung durch den Erker).

Die Methode wurde, mit den gleichen Parametern wie oben bereits beschrieben, an einer weiteren Fassade getestet. An dieser gab es keine Erker, jedoch Verschattungen durch Bäume (Abbildung 13). Punkte in Magenta markieren alle vom Algorithmus erkennbaren Fenster (unter der Annahme, dass es keine Verschattung gäbe). Schwarze Punkte in Abbildung 13 (oben) markieren die richtig detektierten Fenster, rote Punkte markieren Fenster, die nicht erkannt wurden und das schwarze Rechteck markiert eine grobe Fehldetektion (vgl. Abbildung 13 oben und unten).

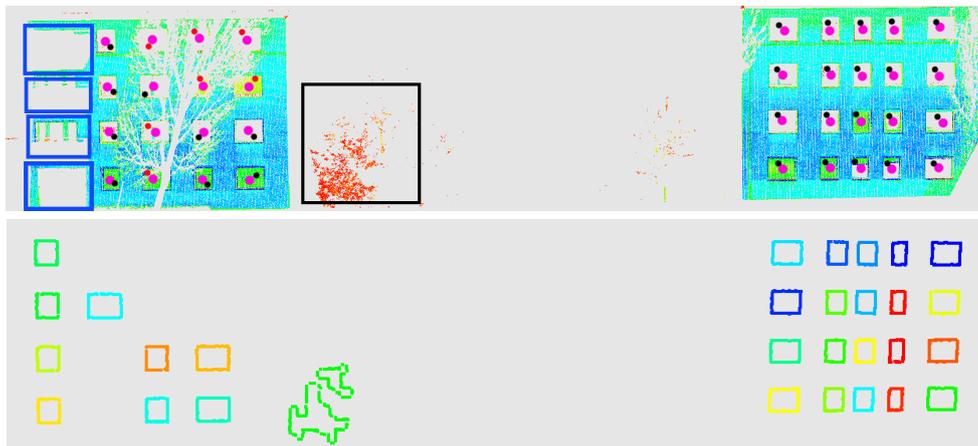


Abb. 13: Das obere Bild zeigt eine Fassade, die durch eine Straße in der Mitte getrennt wurde. Die Detektion der Ebene hat trotz der großen Lücke reibungslos funktioniert. Das untere Bild zeigt das Ergebnis der Fensterdetektion. Siehe Text für eine detaillierte Beschreibung.

Die quantitative Auswertung für diese Fassade zeigt ähnliche Ergebnisse wie für die erste (Tab. 2). Bei diesen Ergebnissen ist klar ersichtlich, dass die Methode bei Fassaden ohne Erker sehr gute Ergebnisse erzielt, jedoch sind Verschattungen, wie bei vielen anderen Methoden ein Problem für die Auswertung. Die blauen Rechtecke in Abbildung 13 (oben) sind Balkons (die oberen drei) und ein Garagentor (unteres Rechteck), welche korrekt als „kein Fenster“ erkannt wurden.

Tab 2: Quantitative Auswertung für die Klassifikation der Fenster für Fassade 2. Treffgenauigkeit und F1-score liegen jeweils bei 80 % und 88 %.

		Tatsächlich	
		Fenster	kein Fenster
Prädiziert	Fenster	TP=29	FP=1
	kein Fenster	FN=7	TN=4

## 5 Fazit & Ausblick

Die hier präsentierte Methode nutzt ein nachvollziehbares, simples Konzept zur Herausarbeitung der Fenster in Fassaden in hochauflösenden 3D Punktwolken. Das Verfahren nutzt die hohe Punktdichte zur groben Bestimmung der Fensterkonturen. Eine Generalisierung der Fensterformen wird vorgenommen, indem für jedes Fenster ein kleinstes umschreibendes Rechteck bestimmt wird. Von diesem Rechteck kann die Fläche robust bestimmt werden und somit die generalisierte Fensterfläche. Die Fensterposition kann z.B. über den Schwerpunkt der Punkte bestimmt werden. In zukünftigen Arbeiten können diese Merkmale (Fläche, Schwerpunkt) dafür verwendet werden, die Stockwerksanzahl und das Fenster-Fassaden-Verhältnis zu bestimmen. Ergebnisse zeigen, dass eine Trefferquote zwischen 80 und 90 % erzielt werden kann. Dies steht allerdings unter der Voraussetzung, dass eine Fassade keine Erker hat oder diese bei der Auswertung nicht berücksichtigt werden. Die Detektion von Fenstern in Erkern ist eine Einschränkung dieser Methode, die aber in zukünftigen, weiterführenden Arbeiten behoben werden kann. Ein möglicher Ansatz hierfür wäre eine weitere Ebenensuche in einem gewissen Abstand vor der bereits detektierten Fassade.

Die weitere Arbeit wird sich damit befassen, das Fenster-Fassaden-Verhältnis und die Stockwerksanzahlen zu bestimmen, da dies wichtige Kenngrößen für energetische Simulationen sind. Vorläufige Ergebnisse zeigen, dass die präzise Berechnung der Fassadenfläche ein offensichtlich wichtiger Schritt für genaue Ergebnisse ist. Das getestete Verfahren (Greedy-Triangulation) funktioniert zwar für unorganisierte Punktwolken meistens gut, verursacht jedoch teils große Lücken im generierten Gitter (Mesh), was zu Fehlern bei der Flächenberechnung führt. Hierfür werden andere Ansätze getestet.

## 6 Dank

Die in dieser Arbeit verwendeten Punktwolkendaten sind im Rahmen des von der Bundesanstalt für Landwirtschaft und Ernährung (BLE) geförderten Projektes “WeBest – Wärmebedarfsprognose von Gebäuden und Stadtquartieren“ entstanden. Die Autoren bedanken sich bei der Stadt Essen für die gute Kooperation während des Projekts. Das Projekt „i\_city: Intelligente Stadt“ wird vom Bundesministerium für Bildung und Forschung (BMBF) unter dem Förderkennzeichen 13FH9I011A gefördert und vom Projektträger VDI Technologiezentrum GmbH für das BMBF betreut. Diese Arbeit ist im Rahmen des „i\_city“ Projekts entstanden.

## 7 Literaturverzeichnis

- ALI, H., SEIFERT, C., JINDAL, N., PALETTA, L. & PAAR, G., 2007: Window Detection in Facades. 14th International Conference on Image Analysis and Processing (ICIAP 2007), 837-842.
- BARBER, C.B., DOBKIN, D.P. & HUHDANPAA, H., 1996: The Quickhull algorithm for convex hulls. ACM Transactions on Mathematical Software, **22**(4), 469-483.
- BILJECKI, F., STOTER, J., LEDOUX, H., ZLATANOVA, S. & ÇÖLTEKIN, A., 2015: Applications of 3D City Models: State of the Art Review. ISPRS International Journal of Geo-Information, **4**(4), 2842-2889.

- BMW, 2011: Research for an environmentally sound, reliable and affordable energy supply. [https://www.bmwi.de/Redaktion/EN/Publikationen/research-for-an-environmentally-sound-reliable-and-affordable-energy-supply.pdf?\\_\\_blob=publicationFile&v=3](https://www.bmwi.de/Redaktion/EN/Publikationen/research-for-an-environmentally-sound-reliable-and-affordable-energy-supply.pdf?__blob=publicationFile&v=3), letzter Zugriff 11.01.2018.
- BMW, 2014: Hoher Energieverbrauch des Gebäudesektors. <https://www.bmwi-energiewende.de/EWD/Redaktion/Newsletter/2014/22/Meldung/hoher-energieverbrauch-des-gebaeudesektor.html>, letzter Zugriff 11.01.2018.
- MONIEN, D., STRZALKA, A., KOUKOFIKIS, A., COORS, V. & EICKER, U., 2017: Comparison of building modelling assumptions and methods for urban scale heat demand forecasting. *Future Cities and Environment*, **3**(1), 2.
- MONIEN, D., WILTING, R., CASPER, E., BRENNENSTUHL, M. & COORS, V., 2016: WeBest – Automatisierte Korrektur und Mapping von Fassadenthermographien auf 3D-Gebäudemodelle. *Photogrammetrie - Fernerkundung - Geoinformation*, **2016**(4), 246-257.
- NEUHAUSEN, M., OBEL, M., MARTIN, A., MARK, P. & KÖNIG, M., 2018: Window detection in facade images for risk assessment in tunneling. *Visualization in Engineering*, **6**(1), 1.
- NGUATEM, W., DRAUSCHKE, M. & MAYER, H., 2014: Localization of Windows and Doors in 3d Point Clouds of Facades. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-3, 87-94.
- NOUVEL, R., BRASSEL, K.-H., BRUSE, M., DUMINIL, E., COORS, V., EICKER, U. & ROBINSON, D., 2015: SimStadt, a new workflow-driven urban energy simulation platform for CityGML city models. J.-L. Scartezzini (Ed.), *Proceedings of International Conference CISBAT 2015 “Future Buildings and Districts – Sustainability from Nano to Urban Scale”*, EPFL Lausanne, LESO-PB, 889-894.
- NOUVEL, R., ZIRAK, M., COORS, V. & EICKER, U., 2017: The influence of data quality on urban heating demand modeling using 3D city models. *Computers, Environment and Urban Systems*, **64**, 68-80.
- RUSU, R.B., 2009: *Semantic 3D Object Maps for Everyday Manipulation in Human Living Environments*. Computer Science department, Technische Universität München.
- RUSU, R.B. & COUSINS, S., 2011: 3D is here: Point Cloud Library (PCL). *IEEE International Conference on Robotics and Automation*, 1-4.
- TRINH, H.-H., KIM, D.-N., KANG, S.-J. & JO, K.-H., 2009: Window Extraction Using Geometrical Characteristics of Building Surface. D.-S. Huang, K.-H. Jo, H.-H. Lee, H.-J. Kang & V. Bevilacqua (Eds.), *Emerging Intelligent Computing Technology and Applications*, Springer Berlin Heidelberg, 585-594.
- TUTTAS, S. & STILLA, U., 2013: Window Detection in Sparse Point Clouds Using Indoor Points. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, **38**(3), 131-136.
- ZHANG, R. & ZAKHOR, A., 2014: Automatic identification of window regions on indoor point clouds using LiDAR and cameras. *IEEE Winter Conference on Applications of Computer Vision*, 107-114.