

**TECHNISCHE
UNIVERSITÄT
DRESDEN**

Fakultät Elektrotechnik und Informationstechnik

Institut für Automatisierungstechnik

DIPLOMARBEIT

zum Thema

Strukturbasierte Multi-View-Erkennung von 3D Objekten mit
traditionellen und Deep-Learning Methoden

vorgelegt von Mo Li
im Studiengang AMR, Jg. 2013
geboren am in China

zur Erlangung des akademischen Grades eines
Diplomingenieurs
(Dipl.-Ing.)

Betreuer: Dipl.-Ing. Fabio Bracci
Dr.-Ing. Zoltán-Csaba Márton
Verantwortlicher Hochschullehrer: Prof. Dr. techn. Klaus Janschek
Tag der Einreichung: 21.12.2017



Aufgabenstellung für die Diplomarbeit

Für: Frau Mo Li

Studiengang: Automatisierungs-, Mess- und Regelungstechnik

Thema: **Strukturbasierte Multi-View-Erkennung von 3D Objekten mit traditionellen und Deep-Learning Methoden**

Die traditionelle und Deep-Learning 3D-Objekterkennung verfolgt den Ansatz der Bestimmung des globalen Deskriptors und der darauf basierenden Klassifizierung. Die Idee für diese Diplomarbeit ist, die geometrischen Beziehungen der Deskriptoren durch die Erstellung eines Deskriptor-Baums zu kodieren, und eine spezifische Matching-Methode für solche strukturierten Daten anzuwenden.

Der Deskriptor-Baum stellt die hierarchische Zusammensetzung von glatten Objektteilen dar, so ähnlich wie die "geometric icons (geons)" der Gestaltpsychologie-Theorie in Biedermanns „Recognition-by-components theory of human perception". Im Deskriptor-Baum können die Deskriptoren entweder traditionell (VFH) oder Deep-learning (Autoencoder) Deskriptoren sein. In der Arbeit wird eine Untermenge der SHREC 3D-Objektdatenbank zur Auswertung verwendet, und es werden mehrere 2.5D-Ansichten der Objekte betrachtet.

Alle Ergebnisse dieser Arbeit sind vollständig und nachvollziehbar zu dokumentieren. Ergebnisse aus Fremdquellen müssen nach den üblichen Zitierregeln eindeutig gekennzeichnet werden.

Im Rahmen der Arbeit sollen folgende Teilaufgaben bearbeitet werden:

1. Einarbeitung in die verwendete Software
2. Strukturierte Anforderungsdefinition
3. Überblick über den aufgaben-bezogenen Stand der Wissenschaft und Technik sowie Abgrenzung des eigenen Beitrags
4. Erfassung einer "Baseline" zur späteren Auswertung des eigenen Beitrags
5. Erfassung einer vorbereitenden Analyse der Zerteilung eines 3D Objektes
6. Entwurf einer strukturbasierten Repräsentation eines 3D Objektes
7. Implementierung der in Teilaufgabe 6 entworfenen Deskriptor im Rahmen der PCL Library
8. Test und Auswertung der entwickelten Repräsentation mittels Daten von der SHREC 2010 Datensatz und die Baseline aus Teilaufgabe 4
9. Dokumentation der Ergebnisse

Betreuer: Fabio Bracci, DLR

Betreuer: Dr. Zoltán-Csaba Márton, DLR

Ausgehändigt: 01.01.2018

Einzureichen: 10.06.2018

Prof. Dr. techn. K. Janschek
Verantwortlicher Hochschullehrer

Inhaltsverzeichnis

1	Einleitung	1
1.1	Aufgabestellung	2
2	Anforderungsdefinition	3
2.1	Strukturierte Analyse	3
2.1.1	Kontextdiagramm	3
2.1.2	Datenflussdiagramm - Ebene 1	4
2.1.3	Datenflussdiagramm - Ebene 2 - Funktion 1	5
2.1.4	Datenflussdiagramm - Ebene 2 - Funktion 3	5
2.1.5	Datenflussdiagramm - Ebene 2 - Funktion 4	6
Anhang A	Abkürzungsverzeichnis	11
Anhang B	Literaturverzeichnis	13

Abbildungsverzeichnis

2.1	Kontextdiagramm	3
2.2	Datenflussdiagramm - Ebene 1	4
2.3	Datenflussdiagramm - Ebene 2 - Funktion 1	5
2.4	Datenflussdiagramm - Ebene 2 - Funktion 3	7
2.5	Datenflussdiagramm - Ebene 2 - Funktion 4	8

Tabellenverzeichnis

2.1	Kontextdiagramm - Daten- und Steuerflüsse	4
2.2	Datenflussdiagramm - Ebene 1 - Funktionen	5
2.3	Datenflussdiagramm - Ebene 1 - Datenflüsse	6
2.4	Datenflussdiagramm - Ebene 2 - F1 - Funktionen	6
2.5	Datenflussdiagramm - Ebene 2 - F1 - Datenflüsse	7
2.6	Datenflussdiagramm - Ebene 2 -F3 - Funktionen	7
2.7	Datenflussdiagramm - Ebene 2 - F4 - Funktionen	8

Quelltextverzeichnis

1 Einleitung

In den meisten Roboteranwendungen ist die Objekterkennung und Klassifizieren eine der wichtigsten Aufgaben. Im Vergleich zur 2D-Bildverarbeitung hat 3D-Objekterkennung mehrere Vorteilen und rückt als Forschungsthema mehr in den Fokus, da es Lichtbedingungen, Farbveränderungen, radiale Verzerrungen usw, welche die Erkennung in 2D beeinflussen, vermeidet.

Um 3D Objekte zu erkennen, gibt es 2 Methoden: die Erste ist Matching durch Lokale Deskriptoren; die zweite Methode ist das Matching durch globalen Deskriptoren. Dieses Thema behandelt hauptsächlich globale Deskriptoren. Globale Deskriptoren codieren die Objektgeometrie. Sie werden nicht für einzelne Punkte berechnet, sondern für einen ganzen Cluster, der ein Objekt darstellt. Globale Deskriptoren werden zur Objekterkennung und Klassifizierung, zur geometrischen Analyse (Objektyp, Form) und zur Posenschätzung verwendet. Viewpoint Feature Histogram(VFH)[4], Clustered Viewpoint Feature Histogram(CVFH)[5] und Oriented, Unique and Repeatable CVFH (OURCVFH)[2] sind 3 gebräuchlich globalen Deskriptoren, die für PointsCloud entwickelt werden.

Deep Learning ist jetzt sehr populär und erfolgreich. Ein wichtiger Grund ist, dass Deep Learning mit Rohdaten beginnen kann, da die Deskriptor beim Lernen automatisch vom neuronalen Netzwerk erstellt werden und die Zielfunktion aus den Daten gelernt werden kann. Deep Learning verschiebt die Last des Feature-Designs zu dem zugrundeliegenden Lernsystem und Klassifikationslernensystem, das typisch für ein früheres mehrschichtiges neuronales Netzwerklernen ist. Beim DLR in Institut RMC werden Variation-Auto-Encoder (VAE) als eine unbeaufsichtigte Deep-Learning Klassifikatorentrainingsmethode eingesetzt.

Open Source, Point Cloud Libray(PCL)[1], ist ein 3D-Bilderarbeitung Tool und hat inzwischen einen ähnlichen Status wie OpenCV für die 2D-Bildverarbeitung erreicht. PCL enthält zahlreichen Algorithmen zur Verarbeitung n-dimensionaler Punktwolken und 3D Geometrien. Dieser Deskriptor-Tree wird mithilfe der Software, PCL , realisiert.

1.1 Aufgabestellung

In CVFH und OURCVFH wird das Objekt in kontinuierliche Regionen unterteilt. In einem Objekt können mehreren Deskriptoren erzeugt werden. In der aktuellen Implementierung in PCL sind die extrahierten Histogramm Features in einer ungeordneten Liste.

Das Ziel dieser Arbeit ist es einen Deskriptor-Baum zu bauen, wodurch die geometrischen Beziehungen der Deskriptoren strukturiert abgelegt werden können. Stand der Technik ist es, dass mithilfe von Nächste-Nachbarn-Klassifikation[3] die VFH, CVFH und OURCVFH Deskriptoren geschätzt und klassifiziert werden. Wird ein 3D Baum als Ansatz gewählt, so können die Deskriptoren durch Berechnung der Entfernung der Bäume[6] klassifiziert werden.

Ein Problem beim Deep-Learning ist, dass End-to-End DeepLearning für 3D-Objekte möglich ist, aber kompliziert, da die Oberflächen mit Hilfe von Differentialgeometrie und komplizierten Windungen betrachtet werden müssen. Daher werden wir versuchen, die Struktur anders zu betrachten.

2 Anforderungsdefinition

Die Anforderungsspezifikation steht am Anfang des Entwicklungsprozesses. Sie bildet die nötige Grundlage für die Entwicklung eines technischen Systems und ist von großer Bedeutung für den Erfolg dieser Entwicklung.

2.1 Strukturierte Analyse

2.1.1 Kontextdiagramm

Ein Kontextdiagramm dient der Modellierung einer Systemumgebung in einer frühen Entwurfs- oder Analysephase. Das Kontextdiagramm stellt die oberste Hierarchieebene von Datenflussdiagrammen dar. Es handelt sich um ein abstraktes Datenflussdiagramm, mit dem die Schnittstellen des Systems zu dessen Umwelt abgebildet werden. Im folgenden Daten- und Steuerkontextdiagramm dieser Arbeit wird der Eingang und Ausgang der Schnittstelle für die Ansteuerung des youBot-Systems bezeichnet, siehe Abbildung 2.1. Die Tabelle 2.1 beschreibt die angeforderten Datenflüsse.

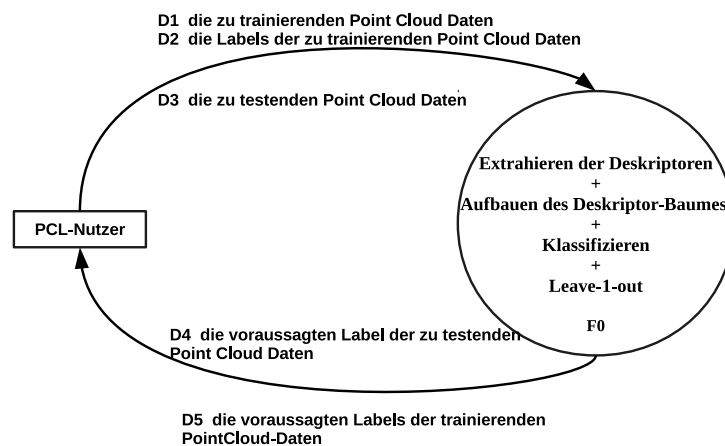


Abbildung 2.1: Kontextdiagramm

Tabelle 2.1: Kontextdiagramm - Daten- und Steuerflüsse

Datenfluss	Beschreibung
D1 die zu trainierenden Point Cloud Daten	Die Trainingsdaten im PCD Form, um gute Klassifikatoren zu lernen
D2 die Labels der zu trainierenden Point Cloud Daten	die Labels für die Trainingsdaten
D3 die zu testenden Point Cloud Daten	Die Testdaten in PCD Form, um die gelernten Klassifikatoren zu testen
D4 die Labels der getesteten Point Cloud Daten	die Labels für die getesteten Daten , die bei den gelernten Klassifikatoren vorausgesagt werden
D5 die Labels der trainierten Point Cloud Daten	die Labels der Trainingsdaten, die bei den gelernten klassifikatoren vorausgesagt werden,

2.1.2 Datenflussdiagramm - Ebene 1

Die Aufgabe wird in mehrere Teilaufgaben zerlegt, die Abbildung 2.2 zeigt die Komposition auf Ebene 1. Die Tabellen 2.2 und 2.3 beschreiben die enthaltenen Funktionen und Datenflüsse.

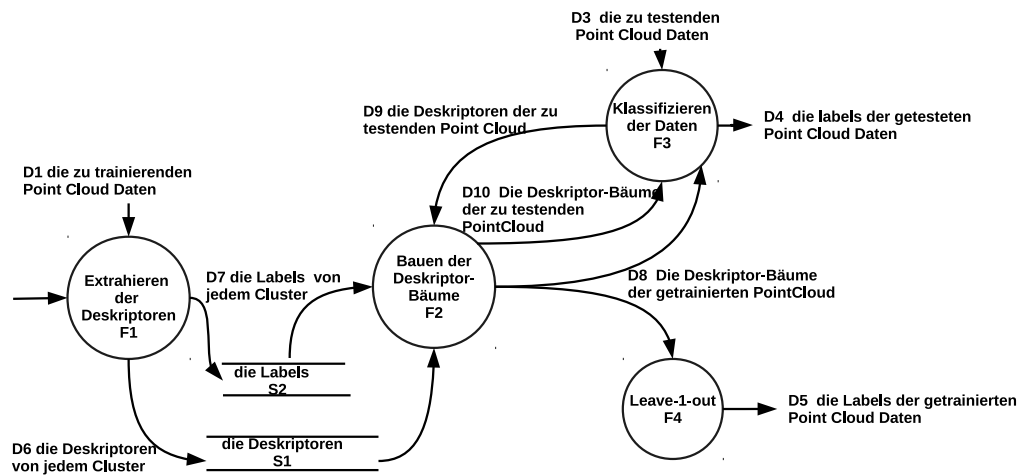


Abbildung 2.2: Datenflussdiagramm - Ebene 1

Tabelle 2.2: Datenflussdiagramm - Ebene 1 - Funktionen

Funktion	Beschreibung
F1 Extrahieren der Deskriptoren	Das Objekt wird in mehrere Cluster unterteilt und die Deskriptoren von jedem Cluster werden berechnet.
F2 Bauen des Dekriptor-Baumes	Der Dekriptor-Baum wird gebaut
F3 Klassifizieren der Daten	Die Testdaten werden nach den gelernten Klassifikatoren klassifiziert und erhalten die Prognoselabel
F4 Leave-1-out	die Trainingdaten werden nach den gelernten Klassifikatoren klassifizieren und erhalten die Prognoselabel

2.1.3 Datenflussdiagramm - Ebene 2 - Funktion 1

Die im Abschnitt 2.2.2 dargestellten Funktion F1 wird weiter detailliert, um das Teilsystem deutlich zu machen.

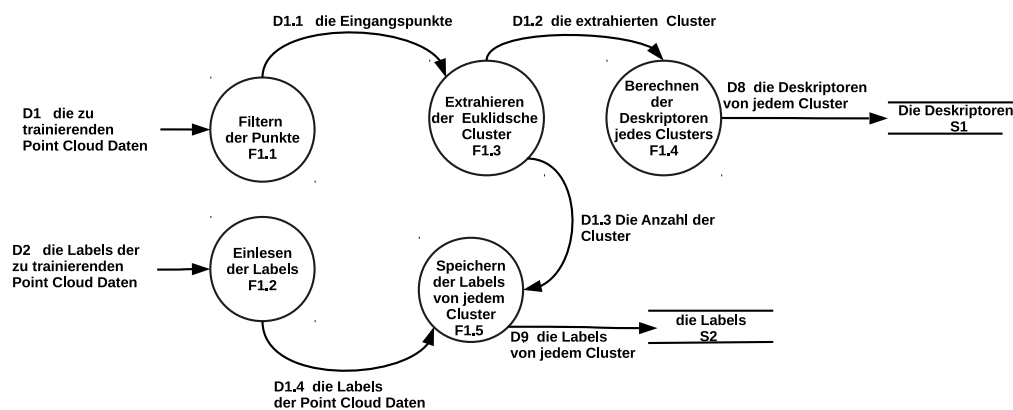


Abbildung 2.3: Datenflussdiagramm - Ebene 2 - Funktion 1

2.1.4 Datenflussdiagramm - Ebene 2 - Funktion 3

Die Abbildung 2.4 stellt die Dekomposition der Funktion: F3 dar. Die Tabellen 2.8 und 2.9 beschreiben die enthaltenen Subfunktionen und Datenflüsse.

Tabelle 2.3: Datenflussdiagramm - Ebene 1 - Datenflüsse

Datenfluss	Beschreibung
D6 die Deskriptoren von jedem Cluster	die Deskriptoren für jeden untergeteilten Cluster
D7 die Labels von jedem Cluster	die Labels für jeden erzeugten Cluster
D8 Die Deskriptor-Bäume der getrainierten PointCloud	Die Deskriptor-Bäume von allen Trainingsobjekt
D9 die Deskriptoren der zu testenden Point Cloud	die Deskriptoren für die Test Objekten
D10 Die Deskriptor-Bäume der zu testenden PointCloud	Die Deskriptor-Bäume für die Test Objekten

Tabelle 2.4: Datenflussdiagramm - Ebene 2 - F1 - Funktionen

Funktion	Beschreibung
F1.1 Filtern der Punkte	Die Punkte, die mit hohen Krümmung, sind gefiltert
F1.2 Einlesen der Labels	Die Labels der Point Clouds werden heruntergeladet.
F1.3 Extrahieren der Euklidische Cluster	Das Objekt wird mit bestimmten Euklidischen Distanz unterteilt
F1.4 Berechnen der Deskriptoren jedes Clusters	Die Deskriptoren werden nach den unterteilten Clustern berechnet
F1.5 Speichern der Labels von jedem Cluster	Entsprechend der Nummer werden die Labels gespeichert

2.1.5 Datenflussdiagramm - Ebene 2 - Funktion 4

Tabelle 2.5: Datenflussdiagramm - Ebene 2 - F1 - Datenflüsse

Daten- oder Steuerfluss	Beschreibung
D1.1 die Einganspunkte	die Punkte in der PointCloud nach dem Filtern
D1.2 die extrahierten Cluster	die unterteilten Cluster
D1.3 Die Anzahl der Cluster	Die Anzahl der Cluster der existierenden Cluster
D1.4 die Labels der Point Cloud Daten	Die Labels von jeder PointCloud

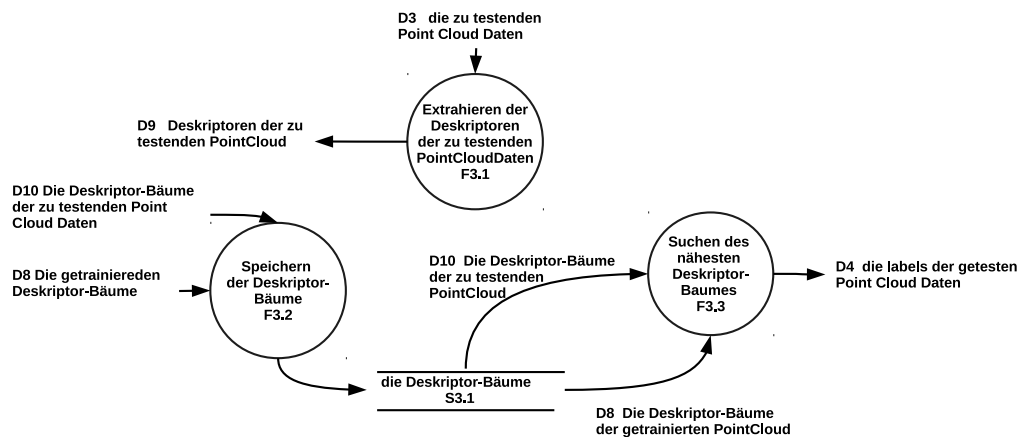

Abbildung 2.4: Datenflussdiagramm - Ebene 2 - Funktion 3

Tabelle 2.6: Datenflussdiagramm - Ebene 2 -F3 - Funktionen

Funktion	Beschreibung
F3.1 Extrahieren der Deskriptoren der zu testenden PointCloudDaten	Die Deskriptoren der Point-Cloud werden als Testdaten berechnet.
F3.2 Speichern der Deskriptor-Bäume	Die erzeugten Deskriptorbäume werden gespeichert
F3.3 Suchen des nächsten Deskriptor-Baumes	Die nächsten Deskriptorbäume werden in dem gespeicherten und trainierten Baum gesucht.

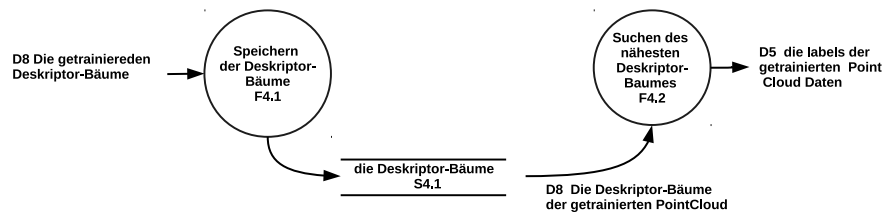


Abbildung 2.5: Datenflussdiagramm - Ebene 2 - Funktion 4

Tabelle 2.7: Datenflussdiagramm - Ebene 2 - F4 - Funktionen

Funktion	Beschreibung
F4.1 Speichern der Deskriptor-Bäume	Die erzeugten Deskriptorbäume werden gespeichert
F4.2 Suchen des nächsten Deskriptor-Baumes	Die nächsten Deskriptorbäume werden in dem gespeicherten und trainierten Baum gesucht.

Anhang

A Literaturverzeichnis

- [1] PCL - Point Cloud Library(PCL). [Online]: <http://pointclouds.org/>
- [2] Aldoma, A., Tombari, F., Rusu, R.B., Vincze, M., M.: OUR-CVFH – Oriented, Unique and Repeatable Clustered Viewpoint Feature Histogram for Object Recognition and 6DOF Pose Estimation.
- [3] Cover, Thomas, and Peter Hart. "Nearest neighbor pattern classification." *Information Theory, IEEE Transactions on* 13.1 (1967): 21-27.
- [4] Rusu, R.B.; Bradski, G.; Thibaux, R.; Hsu, J., "Fast 3D recognition and pose using the Viewpoint Feature Histogram," *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on* , vol., no., pp.2155,2162, 18-22 Oct. 2010.
- [5] Aldoma, A., Blodow, N., Gossow, D., Gedikli, S., Rusu, R.B., Vincze, M., Bradski, G.: CAD-Model Recognition and 6DOF Pose Estimation Using 3D Cues. In: *3DRR Workshop, ICCV (2011)*
- [6] Torsello, A., Hidovic, D., Pelillo, M.: Polynomial-time metrics for attributed trees. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 27(7), 1087-1099 (Jul 2005)

Selbstständigkeitserklärung

Hiermit versichere ich, Mo Li, geboren am in China, dass ich die vorliegende Diplomarbeit zum Thema

*Strukturbasierte Multi-View-Erkennung von 3D Objekten mit
traditionellen und Deep-Learning Methoden*

ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe; die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht. Bei der Auswahl und Auswertung des Materials sowie bei der Herstellung des Manuskripts habe ich Unterstützungsleistungen von folgenden Personen erhalten:

Dipl.-Ing. Fabio Bracci, Dr.-Ing. Zoltán-Csaba Márton

Weitere Personen waren an der geistigen Herstellung der vorliegenden Diplomarbeit nicht beteiligt. Mir ist bekannt, dass die Nichteinhaltung dieser Erklärung zum nachträglichen Entzug des Diplomabschlusses (Masterabschlusses) führen kann.

Dresden, den 21.12.2017

.....
Unterschrift