Proseminar Objektposenschätzung

Robert Jeutter

11. November 2021

Damit ein Roboter einen Gegenstand greifen kann, ist es meist notwendig die genaue Lage des Objektes zu kennen. Dies kann sowohl über klassische Verfahren als auch über Deep-Learning-Verfahren erreicht werden. Ziel dieses Seminars ist es den Stand der Technik für die Objektposenschätzung aufzuarbeiten und vorzustellen. Der Fokus sollte dabei auf Verfahren liegen, bei denen zuvor kein Objektmodell benötig wird, sodass auch die Lage von unbekannten Objekten geschätzt werden kann.

1 Motivation

Die Erkennung von Objekten und die Schätzung ihrer Lage in 3D hat eine Vielzahl von Anwendungen in der Robotik. So ist beispielsweise die Erkennung der 3D-Lage und Ausrichtung von Objekten wichtig für die Robotermanipulation. Sie ist auch bei Aufgaben der Mensch-Roboter-Interaktion nützlich, z. B. beim Lernen aus Demonstrationen.
Traditionell wird das Problem der Objektposenschätzung durch den Abgleich von Merkmalspunkten zwischen 3D-Modellen und Bildern angegangen. Diese Methoden setzen jedoch voraus, dass die Objekte reichhaltig texturiert sind, um Merkmalspunkte für den Abgleich zu erkennen. Daher sind sie nicht in der Lage, mit Objekten ohne Textur umzugehen. Die meisten bestehenden Ansätze zur Objektposenschätzung setzen den Zugriff auf das 3D-Modell einer Objektinstanz voraus. Der Zugang zu solchen 3D-Modellen erschwert die Verallgemeinerung auf neue, unbekannte Instanzen. Darüber hinaus erfordern 3D-Modelldatenbanken oft einen nicht unerheblichen manuellen Aufwand und Expertenwissen, um sie zu erstellen, wobei Schritte wie Scannen, Netzverfeinerung oder CAD-Design erforderlich sind. Unordnung und Verdeckungen zwischen den Objekten senken die korrekte Erkennung bei modellbasierten Verfahren zudem deutlich. Bei schablonenbasierten Methoden wird eine starre Schablone konstruiert und verwendet, um verschiedene Stellen im Eingabebild zu scannen. An jeder Stelle wird ein Ähnlichkeitswert berechnet, und die beste Übereinstimmung wird durch den Vergleich dieser Ähnlichkeitswerte ermittelt. Schablonenbasierte Methoden sind nützlich für die Erkennung texturloser Objekte. Sie können jedoch nicht sehr gut mit Verdeckungen zwischen Objekten umgehen, da die Vorlage einen niedrigen Ähnlichkeitswert hat, wenn das Objekt verdeckt ist. Alternativ dazu können Methoden, die eine Regression von Bildpixeln auf 3D-Objektkoordinaten erlernen, um 2D-3D-Korrespondenzen für die 6D-Positionsschätzung herzustellen, nicht mit symmetrischen Objekten umgehen. Außerdem können bei der Verfolgung durch solche dynamische on-the-fly Rekonstruktion von Objekten Fehler entstehen, wenn Beobachtungen mit fehlerhaften Posenschätzungen in das globale Modell einfließen. Diese Fehler wirken sich nachteilig auf die Modellverfolgung in nachfolgenden Bildern aus.

2 Kategorisierung

Motiviert durch die oben genannten Einschränkungen, zielt diese Arbeit auf eine genaue, robuste 6D-Objekterkennung ab, die auf neuartige Objekte ohne 3D-Modelle verallgemeinert werden kann. Die Kategorisierung aller Verfahren erfolgt nach folgendem Schemata

Modell Müssen für das Training oder Nutzung merkmalsbasierte oder Objektmodelle (2D, 3D, CAD) vorhanden sein?

Video-Input Verarbeitet das Verfahren 2D Bilder, 3D Bilder mit Tiefenwahrnehmung?

Datensatz mit welchen Datensätzen wurde das Verfahren trainiert oder getestet?

Genauigkeit Wie akkurat ist die Objektposenschätzung im Vergleich?

Ressourcen Wie Ressourcenintensiv ist das Verfahren? Wird spezielle Hardware benötigt?

Laufzeit Mit welcher Geschwindigkeit ist die Verarbeitung von Eingabedaten möglich und stabil?

3 Verschiedene Verfahren

3.1 BundleTrack[BundleTrack]

Modell
Video-Input
Datensatz
Genauigkeit
Ressourcen
Laufzeit

3.2 DeepIM[Deepim]

Modell
Video-Input
Datensatz
Genauigkeit
Ressourcen
Laufzeit

3.3 MaskFusion[MaskFusion]

Modell
Video-Input
Datensatz
Genauigkeit
Ressourcen
Laufzeit

3.4 Neural Analysis-by-Synthesis[CategoryLevelObject]

Modell
Video-Input
Datensatz
Genauigkeit
Ressourcen
Laufzeit

3.5 6-PACK[6pack]

Modell
Video-Input
Datensatz
Genauigkeit
Ressourcen
Laufzeit

3.6 PoseCNN[PoseCNN]

Modell
Video-Input
Datensatz
Genauigkeit
Ressourcen
Laufzeit

3.7 Robust Gaussian Filter[GaussianFilter]

Modell
Video-Input
Datensatz
Genauigkeit
Ressourcen
Laufzeit

4 Vergleich verschiedener Verfahren

5 Fazit

Glossar

Convolutional Neural Network Besitzt pro Convolutional Layer mehrere Filterkerne, sodass Schichten an Feature Maps entstehen, die jeweils die gleiche Eingabe bekommen, jedoch aufgrund unterschiedlicher Gewichtsmatrizen unterschiedliche Features extrahieren.. 2

Quaternion-Darstellung Darstellung V einer Gruppe G, die einen G-invarianten Homomorphismus $J : V \to V$ besitzt, der antilinear ist und $J^2 = -Id$ erfüllt.. 3