



(10) **DE 10 2022 201 719 A1** 2023.08.24

(12) **Offenlegungsschrift**

(21) Aktenzeichen: **10 2022 201 719.6**

(22) Anmeldetag: **18.02.2022**

(43) Offenlegungstag: **24.08.2023**

(51) Int Cl.: **G06V 10/70 (2022.01)**
G06N 20/00 (2019.01)

(71) Anmelder:

**Robert Bosch Gesellschaft mit beschränkter
Haftung, 70469 Stuttgart, DE**

(72) Erfinder:

**Adrian, David, 70565 Stuttgart, DE; Graf,
Christian, 81371 München, DE; Kupcsik, Andras
Gabor, 71034 Böblingen, DE**

(56) Ermittelter Stand der Technik:

DE 10 2020 208 211 A1

**CHEN, Boyuan; ABBEEL, Pieter; PATHAK,
Deepak. Unsupervised learning of visual 3d
keypoints for control. In: International
Conference on Machine Learning. PMLR, 2021.
Dokumentenseiten 1-11. [online abrufbar über**

**[http://proceedings.mlr.press/v139/chen21b/
chen21b.pdf](http://proceedings.mlr.press/v139/chen21b/chen21b.pdf)**

**FLORENCE, P. R., Manuelli, L., Tedrake, R.:
Dense object nets: Learning dense visual object
descriptors by and for robotic manipulation. In:
arXiv preprint arXiv:1806.08756, 2018.
Dokumentenseiten 1-12. doi: 10.48550/
arXiv.1806.08756 (von der Anmelderin
selbstgenannter Stand der Technik)**

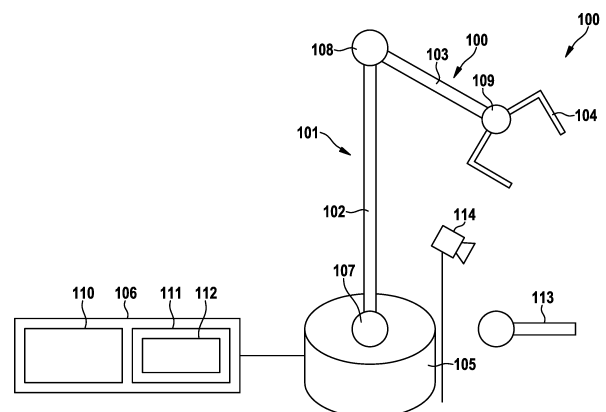
**REUSE, M., Simon, M., Sick, B.: About the
Ambiguity of Data Augmentation for 3D Object
Detection in Autonomous Driving. In:
Proceedings of the IEEE/CVF International
Conference on Computer Vision. 2021. S. 979-
987. [online abrufbar über [https://openaccess.
thecvf.com/content/ICCV2021W/ERCVAD/papers/
Reuse_About_the_Ambiguity_of_Data_Augmen-
tation_for_3D_Object_Detection_ICCVW_2021_-
paper.pdf](https://openaccess.thecvf.com/content/ICCV2021W/ERCVAD/papers/Reuse_About_the_Ambiguity_of_Data_Augmentation_for_3D_Object_Detection_ICCVW_2021_paper.pdf)]**

Rechercheantrag gemäß § 43 PatG ist gestellt.

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen.

(54) Bezeichnung: **Vorrichtung und Verfahren zum Trainieren eines maschinellen Lernmodells zum Erzeugen von Deskriptorbildern zu Bildern von Objekten**

(57) Zusammenfassung: Gemäß verschiedenen Ausführungsformen wird ein Verfahren zum Trainieren eines maschinellen Lernmodells zum Erzeugen von Deskriptorbildern zu Bildern von ein oder mehreren Objekten beschrieben, aufweisend aufnehmen von mehreren Kamerabildern, wobei jedes Kamerabild ein oder mehrere Objekte zeigt, für jedes Kamerabild Erzeugen von ein oder mehreren augmentierten Versionen des Kamerabilds durch Anwenden, für jede augmentierte Version des Kamerabilds, einer jeweiligen Augmentierung auf das Kamerabild, wobei die Augmentierung eine Positionsänderung von Pixelwerten des Kamerabilds aufweist, und Erzeugen Trainingsbildpaaren, die jeweils das Kamerabild und eine Augmentierung des Kamerabilds oder zwei augmentierte Versionen des Kamerabilds aufweisen und Trainieren des maschinellen Lernmodells mittels Kontrast-Verlust unter Verwendung der Trainingsbildpaare wobei Deskriptorwerte, die das maschinelle Lernmodell für einander entsprechende Pixel erzeugt, als Positiv-Paare und Deskriptorwerte, die das maschinelle Lernmodell für einander nicht entsprechende Pixel erzeugt, als Negativ-Paare verwendet werden.



Beschreibung

Offenbarung der Erfindung

Stand der Technik

[0001] Die vorliegende Offenbarung bezieht sich auf Vorrichtungen und Verfahren zum Trainieren eines maschinellen Lernmodells zum Erzeugen von Deskriptorbildern zu Bildern von Objekten.

[0002] Um eine flexible Herstellung oder Bearbeitung von Objekten durch einen Roboter zu ermöglichen, ist es wünschenswert, dass der Roboter in der Lage ist, ein Objekt ungeachtet der Stellung, mit der das Objekt in den Arbeitsraum des Roboters platziert wird, zu handhaben. Daher sollte der Roboter fähig sein, zu erkennen, welche Teile des Objektes sich an welchen Positionen befinden, sodass er zum Beispiel das Objekt an der korrekten Stelle greifen kann, um es z. B. an einem anderen Objekt zu befestigen oder zu schweißen. Dies bedeutet, dass der Roboter fähig sein sollte, die Pose (Position und Orientierung) des Objekts zum Beispiel aus einem oder mehreren Bildern, die durch eine am Roboter befestigte Kamera aufgenommen werden, zu erkennen bzw. die Position von Stellen zum Aufnehmen oder Bearbeiten zu ermitteln. Ein Ansatz zum Erzielen davon besteht darin, Deskriptoren, d. h. Punkte (Vektoren) in einem vordefinierten Deskriptorraum, für Teile des Objekts (d. h. in einer Bildebene repräsentierte Pixel des Objekts) zu bestimmen, wobei der Roboter trainiert ist, unabhängig von einer aktuellen Pose des Objekts den gleichen Teilen eines Objekts die gleichen Deskriptoren zuzuweisen und somit die Topologie des Objekts in dem Bild zu erkennen, sodass dann beispielsweise bekannt ist, wo sich welche Ecke des Objekts in dem Bild befindet. Bei Kenntnis der Pose der Kamera lässt sich dann wiederum auf die Pose des Objekts rückschließen. Das Erkennen der Topologie lässt sich mit einem maschinellen Lernmodell realisieren, das entsprechend trainiert wird.

[0003] Ein Beispiel hierfür ist das in der Veröffentlichung „Dense Object Nets: Learning Dense Visual Object Descriptors By and For Robotic Manipulation“ von Peter Florence et al. (im Folgenden als „Referenz 1“ bezeichnet) beschriebene dichte Objektnetz. Das dichte Objektnetz wird hierbei auf selbst-überwachte Weise trainiert, wobei eine dedizierte Anordnung mit einer RGBD-Kamera (RGB plus Tiefe) und einem Roboterarm verwendet wird, um Trainingsdaten zu sammeln. Dabei werden mehrere Bilder aus unterschiedlichen Perspektiven derselben Objektanordnung aufgenommen. Dies ist verhältnismäßig zeitintensiv und verlangt eine aufwändige Anordnung mit beweglicher Kamera. Es sind deshalb effizientere Herangehensweisen zum Trainieren eines dichten Objektnetzes wünschenswert.

[0004] Gemäß verschiedenen Ausführungsformen wird ein Verfahren zum Trainieren eines maschinellen Lernmodells zum Erzeugen von Deskriptorbildern zu Bildern von ein oder mehreren Objekten bereitgestellt, aufweisend Aufnahmen von mehreren Kamerabildern, wobei jedes Kamerabild ein oder mehrere Objekte zeigt. Das Verfahren weist für jedes Kamerabild auf: Erzeugen von ein oder mehreren augmentierten Versionen des Kamerabilds durch Anwenden, für jede augmentierte Version des Kamerabilds, einer jeweiligen Augmentierung auf das Kamerabild, wobei die Augmentierung eine Positionsänderung von Pixelwerten (und optional Farbwertänderung von Pixelwerten) des Kamerabilds aufweist, Erzeugen von ein oder mehreren Trainingsbildpaaren, die jeweils das Kamerabild und eine Augmentierung des Kamerabilds oder zwei augmentierte Versionen des Kamerabilds aufweisen und Ermitteln, für jedes Trainingsbildpaar, gemäß der Positionsänderung von Pixelwerten, die die Augmentierung aufweist, mit der die augmentierte Version erzeugt wurde, falls das Trainingsbildpaar eine augmentierte Version des Kamerabilds aufweist, oder gemäß den Positionsänderungen von Pixelwerten, die die beiden Augmentierungen aufweisen, mit der die augmentierten Versionen erzeugt wurden, falls das Trainingsbildpaar zwei augmentierte Versionen des Kamerabilds aufweist, von einander entsprechenden Pixeln und einander nicht entsprechenden Pixeln des Trainingsbildpaars. Das Verfahren weist außerdem das Trainieren des maschinellen Lernmodells mittels Kontrast-Verlust unter Verwendung der Trainingsbildpaare wobei Deskriptorwerte, die das maschinelle Lernmodell füreinander entsprechende Pixel erzeugt, als Positiv-Paare und Deskriptorwerte, die das maschinelle Lernmodell füreinander nicht entsprechende Pixel erzeugt, als Negativ-Paare verwendet werden, auf.

[0005] Das oben beschriebene Verfahren ermöglicht eine erheblich vereinfachte Trainingsdatensammlung und entsprechend ein erheblich weniger aufwändiges Training eines maschinellen Lernmodells zum Erzeugen von Deskriptorbildern als beispielsweise das Vorgehen von Referenz 1, da z.B. nur eine einzige statische RGB- oder RGB-D-Kamera zum Erzeugen der Trainingsdaten erforderlich ist. Für eine Szene (Objektkonfiguration) braucht nur ein einziges Bild aufgenommen zu werden, wodurch einfacher ein diverser Trainingsdatensatz erzeugt werden kann, der zum Training des maschinellen Lernmodells geeignet ist.

[0006] Im Folgenden werden verschiedene Ausführungsbeispiele angegeben.

[0007] Ausführungsbeispiel 1 ist ein Verfahren zum Steuern eines Roboters, wie oben beschrieben.

[0008] Ausführungsbeispiel 2 ist das Verfahren nach Ausführungsbeispiel 1, wobei die Augmentierung eine Rotation, eine perspektivische Transformation und/oder eine affine Transformation aufweist.

[0009] Solche Augmentierungen ermöglichen die Erzeugung eines reichhaltigen Trainingsdatensatzes und ermöglichen gleichzeitig eine einfache und zuverlässige Ermittlung voneinander entsprechenden Pixeln.

[0010] Ausführungsbeispiel 3 ist das Verfahren nach Ausführungsbeispiel 1 oder 2, aufweisend, vor dem Erzeugen der augmentierten Versionen und der Trainingsbildpaare, Zuschneiden der Kamerabilder unter Berücksichtigung von Objektmasken der ein oder mehreren Objekten.

[0011] Damit wird die Menge der Eingangsdaten für das Training (insbesondere für die Erzeugung der augmentierten Versionen der Kamerabilder reduziert).

[0012] Ausführungsbeispiel 4 ist das Verfahren nach einem der Ausführungsbeispiele 1 bis 3, wobei das maschinelle Lernmodell ein neuronales Netz ist.

[0013] In anderen Worten wird ein dichtes Objekt-Netz trainiert. Mit diesen können gute Resultate zur Erzeugung von Deskriptorbildern erzielt werden.

[0014] Ausführungsbeispiel 5 ist das Verfahren nach einem der Ausführungsbeispiele 1 bis 4, wobei die mehreren Kamerabilder aus derselben Perspektive aufgenommen werden.

[0015] Dies vereinfacht das Aufnehmen der Kamerabilder und es kann beispielsweise eine stationäre, z.B. fest montierte Kamera verwendet werden.

[0016] Ausführungsbeispiel 6 ist ein Verfahren zum Steuern eines Roboters zum Aufnehmen oder Bearbeiten eines Objekts, aufweisend Trainieren eines maschinellen Lernmodells nach einem der Ausführungsbeispiele 1 bis 5, Aufnehmen eines Kamerabilds, das das Objekt in einem aktuellen Steuerszenario zeigt, Zuführen des Kamerabilds zu dem maschinellen Lernmodell zum Erzeugen eines Deskriptorbilds, Ermitteln der Position einer Stelle zum Aufnehmen oder Bearbeiten des Objekts in dem aktuellen Steuerszenario aus dem Deskriptorbild und Steuern des Roboters entsprechend der ermittelten Position.

[0017] Ausführungsbeispiel 7 ist das Verfahren nach Ausführungsbeispiel 6, aufweisend Identifizieren einer Referenzstelle in einem Referenzbild, Ermitteln eines Deskriptors der identifizierten Referenzstelle durch Zuführen des Referenzbilds zu dem maschinellen Lernmodell, Ermitteln der Position der Referenz-

stelle in dem aktuellen Steuerszenario durch Suchen des ermittelten Deskriptors in dem aus dem Kamerabild erzeugten Deskriptorbild und Ermitteln der Position der Stelle zum Aufnehmen oder Bearbeiten des Objekts in dem aktuellen Steuerszenario aus der ermittelten Position der Referenzstelle.

[0018] Ausführungsbeispiel 8 ist eine Steuereinrichtung, die eingerichtet ist, ein Verfahren nach einem der Ausführungsbeispiele 1 bis 7 durchzuführen.

[0019] Ausführungsbeispiel 9 ist ein Computerprogramm mit Befehlen, die, wenn sie durch einen Prozessor ausgeführt werden, bewirken, dass der Prozessor ein Verfahren nach einem der Ausführungsbeispiele 1 bis 7 durchführt.

[0020] Ausführungsbeispiel 10 ist ein computerlesbares Medium, das Befehle speichert, die, wenn sie durch einen Prozessor ausgeführt werden, bewirken, dass der Prozessor ein Verfahren nach einem der Ausführungsbeispiele 1 bis 7 durchführt.

[0021] In den Zeichnungen beziehen sich ähnliche Bezugszeichen im Allgemeinen auf dieselben Teile in den ganzen verschiedenen Ansichten. Die Zeichnungen sind nicht notwendigerweise maßstäblich, wobei die Betonung stattdessen im Allgemeinen auf die Darstellung der Prinzipien der Erfindung gelegt wird. In der folgenden Beschreibung werden verschiedene Aspekte mit Bezug auf die folgenden Zeichnungen beschrieben.

Fig. 1 zeigt einen Roboter.

Fig. 2 zeigt ein Diagramm, das einen Datenfluss zum Trainieren eines maschinellen Lernmodells zum Erzeugen von Deskriptorbildern zu Bildern von ein oder mehreren Objekten gemäß einer Ausführungsform veranschaulicht.

Fig. 3 zeigt zwei Trainingsdatenbilder gemäß einer Ausführungsform.

Fig. 4 zeigt ein Ablaufdiagramm, das ein Verfahren zum Trainieren eines maschinellen Lernmodells zum Erzeugen von Deskriptorbildern zu Bildern von ein oder mehreren Objekten gemäß einer Ausführungsform darstellt.

[0022] Die folgende ausführliche Beschreibung bezieht sich auf die begleitenden Zeichnungen, die zur Erläuterung spezielle Details und Aspekte dieser Offenbarung zeigen, in denen die Erfindung ausgeführt werden kann. Andere Aspekte können verwendet werden und strukturelle, logische und elektrische Änderungen können durchgeführt werden, ohne vom Schutzbereich der Erfindung abzuweichen. Die verschiedenen Aspekte dieser Offenbarung schließen sich nicht notwendigerweise gegenseitig aus, da einige Aspekte dieser Offenbarung mit einem oder mehreren anderen Aspekten dieser Offenbarung

kombiniert werden können, um neue Aspekte zu bilden.

[0023] Im Folgenden werden verschiedene Beispiele genauer beschrieben.

[0024] Fig. 1 zeigt einen Roboter 100.

[0025] Der Roboter 100 beinhaltet einen Roboterarm 101, zum Beispiel einen Industrieroboterarm zum Handhaben oder Montieren eines Arbeitsstücks (oder eines oder mehrerer anderer Objekte). Der Roboterarm 101 beinhaltet Manipulatoren 102, 103, 104 und eine Basis (oder Stütze) 105, mittels der die Manipulatoren 102, 103, 104 gestützt werden. Der Ausdruck „Manipulator“ bezieht sich auf die bewegbaren Bauteile des Roboterarms 101, deren Betätigung eine physische Interaktion mit der Umgebung ermöglicht, um z. B. eine Aufgabe auszuführen. Zur Steuerung beinhaltet der Roboter 100 eine (Roboter-) Steuereinrichtung 106, die zum Implementieren der Interaktion mit der Umgebung gemäß einem Steuerprogramm ausgelegt ist. Das letzte Bauteil 104 (das am weitesten von der Stütze 105 entfernt ist) der Manipulatoren 102, 103, 104 wird auch als der Endeffektor 104 bezeichnet und kann ein oder mehrere Werkzeuge beinhalten, wie etwa einen Schweißbrenner, ein Greifinstrument, ein Lackiergerät oder dergleichen.

[0026] Die anderen Manipulatoren 102, 103 (die sich näher an der Stütze 105 befinden) können eine Positionierungsvorrichtung bilden, sodass, zusammen mit dem Endeffektor 104, der Roboterarm 101 mit dem Endeffektor 104 an seinem Ende bereitgestellt ist. Der Roboterarm 101 ist ein mechanischer Arm, der ähnliche Funktionen wie ein menschlicher Arm bereitstellen kann (möglicherweise mit einem Werkzeug an seinem Ende).

[0027] Der Roboterarm 101 kann Gelenkelemente 107, 108, 109 beinhalten, die die Manipulatoren 102, 103, 104 miteinander und mit der Stütze 105 verbinden. Ein Gelenkelement 107, 108, 109 kann ein oder mehrere Gelenke aufweisen, die jeweils eine drehbare Bewegung (d. h. Drehbewegung) und/oder translatorische Bewegung (d. h. Verlagerung) für assoziierte Manipulatoren relativ zueinander bereitstellen können. Die Bewegung der Manipulatoren 102, 103, 104 kann mittels Aktoren initiiert werden, die durch die Steuereinrichtung 106 gesteuert werden.

[0028] Der Ausdruck „Aktor“ kann als eine Komponente verstanden werden, die als Reaktion auf ihren Antrieb zum Bewirken eines Mechanismus oder Prozesses ausgebildet ist. Der Aktor kann durch die Steuereinrichtung 106 erstellte Anweisungen (die sogenannte Aktivierung) in mechanische Bewegungen implementieren. Der Aktor, z. B. ein elektrome-

chanischer Wandler, kann dazu ausgelegt sein, als Reaktion auf seinen Antrieb elektrische Energie in mechanische Energie umzuwandeln.

[0029] Der Ausdruck „Steuereinrichtung“ kann als ein beliebiger Typ von logikimplementierender Entität verstanden werden, die zum Beispiel eine Schaltung und/oder einen Prozessor beinhalten kann, die/der in der Lage ist, in einem Speicherungsmedium gespeicherte Software, Firmware oder eine Kombination davon auszuführen, und die/der Anweisungen, z. B. zu einem Aktor im vorliegenden Beispiel, ausstellen kann. Die Steuereinrichtung kann zum Beispiel durch Programmcode (z. B. Software) konfiguriert werden, um den Betrieb eines Systems, eines Roboters im vorliegenden Beispiel, zu steuern.

[0030] Im vorliegenden Beispiel beinhaltet die Steuereinrichtung 106 einen oder mehrere Prozessoren 110 und einen Speicher 111, der Code und Daten speichert, basierend auf denen der Prozessor 110 den Roboterarm 101 steuert. Gemäß verschiedenen Ausführungsformen steuert die Steuereinrichtung 106 den Roboterarm 101 auf Basis eines maschinellen Lernmodells 112, das im Speicher 111 gespeichert ist.

[0031] Die Steuereinrichtung 106 verwendet das maschinelle Lernmodell 112 dazu, um die Pose eines Objekts 113 zu ermitteln, das zum Beispiel in einen Arbeitsraum des Roboterarms 101 platziert ist. Abhängigkeit von der ermittelten Pose kann die Steuereinrichtung 106 beispielsweise entscheiden, welche Stelle des Objekts 113 durch den Endeffektor 109 gegriffen (oder anderweitige bearbeitet) werden soll.

[0032] Die Steuereinrichtung 106 ermittelt die Pose unter Verwendung des maschinellen Lernmodells 112 unter Verwendung von ein oder mehreren Kamerabildern des Objekts 113. Der Roboter 100 kann zum Beispiel mit einer oder mehreren Kameras 114 ausgestattet sein, die es ihm ermöglichen, Bilder seines Arbeitsraums aufzunehmen. Eine Kamera 114 ist zum Beispiel an dem Roboterarm 101 befestigt, sodass der Roboter Bilder des Objekts 113 von verschiedenen Perspektiven aus aufnehmen kann, indem er den Roboterarm 101 herumbewegt. Es können aber auch ein oder mehrere fixe Kameras vorgehen sein.

[0033] Das maschinelle Lernmodell 112 ist gemäß verschiedenen Ausführungsformen ein (tiefes) neuronales Netzwerk, das für ein Kamerabild eine Merkmalskarte, z.B. in Form eines Bilds in einem Merkmalsraum, erzeugt, die es ermöglicht, Punkte im (2D) Kamerabild Punkten des (3D) Objekts zuzuordnen.

[0034] Beispielsweise kann das maschinelle Lernmodell 112 trainiert sein, einer bestimmten Ecke des Objekts einen bestimmten (eindeutigen) Merkmalswert (auch als Deskriptorwert bezeichnet) im Merkmalsraum zuzuweisen. Wird dem maschinellen Lernmodell 112 dann ein Kamerabild zugeführt und das maschinelle Lernmodell 112 ordnet einem Punkt des Kamerabilds diesen Merkmalswert zu, so kann man folgern, dass sich an dieser Stelle die Ecke befindet (also an einer Stelle im Raum, deren Projektion auf die Kameraebene dem Punkt im Kamerabild entspricht). Kennt man so die Position mehrerer Punkte des Objekts im Kamerabild, kann man die Pose des Objekts im Raum ermitteln.

[0035] Das maschinelle Lernmodell 112 muss für diese Aufgabe geeignet trainiert werden.

[0036] Ein Beispiel für ein maschinelles Lernmodell 112 zur Objekterkennung ist ein dichtes Objektnetz. Ein dichtes Objektnetz bildet ein Bild (z. B. ein durch die Kamera 114 bereitgestelltes RGB-Bild $I \in \mathbb{R}^{H \times W \times 3}$) auf ein beliebig-dimensionales (Dimension D , z.B. $D = 16$) Deskriptorraumbild (auch bezeichnet als Deskriptorbild) $I_D \in \mathbb{R}^{H \times W \times D}$ ab, d.h. jedem Pixel wird ein Deskriptorwert zugewiesen. Das dichte Objektnetz ist ein neuronales Netz, das unter Verwendung von selbstüberwachtem Lernen trainiert ist, ein Deskriptorraumbild für ein Eingabebild eines Bildes auszugeben. Damit können Bilder bekannter Objekte (oder auch unbekannter Objekte) auf Deskriptorbilder abgebildet werden, die Deskriptoren enthalten, die Stellen auf den Objekten unabhängig von der Perspektive des Bildes identifizieren. Damit erhält man eine konsistente Repräsentation von Objekten, die als Grundlage für eine Steuerung z.B. das Manipulieren von Objekten, verwendet werden kann.

[0037] Ein DON kann selbstüberwacht trainiert werden, indem Kontrast-Lernen (engl. contrastive learning) genutzt wird.

[0038] Bei dem in Referenz 1 beschriebenen selbstüberwachten Training wird eine Szene aus verschiedenen Positionen und Winkeln mit einer registrierten RGB-D-Kamera an einem Roboterarm aufgenommen. Da die Ausrichtung der Kamera bekannt ist, bilden die geometrischen Korrespondenzen zwischen zwei dieser Bilder einen Satz von sich entsprechenden und nicht entsprechenden Pixeln. Die Trainingsdaten bestehen aus diesen Sätzen von übereinstimmenden und nicht übereinstimmenden Pixeln zusammen mit den RGB-Bildern. Es ist eine spezielle Kalibrierung der Kamera erforderlich sowie ein ausgeklügeltes Datenerfassungsverfahren, bei dem die am Roboter montierte Kamera die Szene aus verschiedenen Blickwinkeln erfasst. Die Objektdarstellung kann zum Beispiel zur Erkennung von

Schlüsselpunkten auf einem Objekt verwendet werden.

[0039] Das Sammeln der Trainingsdaten für das Training nach Referenz 1 ist somit verhältnismäßig aufwändig, da eine Anordnung mit einer beweglichen Kamera erforderlich ist und Bilder aus unterschiedlichen Perspektiven aufgenommen werden müssen.

[0040] Gemäß verschiedenen Ausführungsformen wird ein Verfahren zum Trainieren eines maschinellen Lernmodells zum Erzeugen von Deskriptorbildern zu Bildern von ein oder mehreren Objekten (z.B. eines DONs) bereitgestellt, bei dem eine statische (d.h. ortsfeste, z.B. fest montierte) Kamera zum Aufnehmen der Bilder verwendet werden kann, d.h. Trainingsbilder aufgenommen werden, die eine jeweilige Trainingsszene aus einer einzigen Perspektive zeigen. Dabei werden Trainingsdaten für das maschinelle Lernmodell erzeugt, indem aufgenommene Trainingsbilder der Trainingsszene augmentiert werden. Dies geschieht auf zweifache (oder mehrfache) Art, so dass zu jedem aufgenommenen Trainingsbild die Trainingsdaten mehrere augmentierte Versionen des aufgenommenen Trainingsbilds enthalten. Statt über geometrische Korrespondenzen von Bildern aus unterschiedlicher Perspektiven Pixel zu ermitteln, die sich entsprechen bzw. nicht entsprechen (wie in Referenz 1) wird gemäß verschiedenen Ausführungsformen aus Informationen über die verwendeten Augmentationen ermittelt, welche Pixel sich entsprechen. Solche Augmentationen (die dies ermöglichen und deshalb gemäß verschiedenen Ausführungsformen verwendet werden können) sind beispielsweise zufällige Rotationen, perspektivische und affine Transformationen. Im Gegensatz zu einer affinen Transformation ist bei einer perspektivischen Transformation die Parallelität nicht Erhalten. Das Bild wird transformiert, als wenn es aus einem anderen Blickwinkel betrachtet würde. Das bedeutet, dass das Bild virtuell „gekippt“ wird (d.h. die Seitenlängen werden ungleich verändert und der Raum dazwischen interpoliert).

[0041] Fig. 2 zeigt ein Diagramm, dass einen Datenfluss zum Trainieren eines maschinellen Lernmodells 201 zum Erzeugen von Deskriptorbildern zu Bildern von ein oder mehreren Objekten veranschaulicht.

[0042] Eine Kamera 202 (z.B. entsprechend der Kamera 114) nimmt ein Kamerabild 203 eines Objekts 204 auf. Das Kamerabild 203 wird durch eine oder mehrere Augmentierungen 204 verarbeitet. Das Ergebnis ist eine oder mehrere augmentierte Versionen 205 des Kamerabilds, z.B. zwei Kamerabildern aus unterschiedlichen Perspektiven (und unterschiedlich von der des ursprünglichen Kamerabilds). Mit ihnen, zusammen mit dem ursprünglichen Kamerabild 203, wird das maschinelle Lernmodell

201 trainiert, wobei Kamerabilder 203 für verschiedene Szenen und verschiedene Objekte aufgenommen werden können und zusammen mit ihren augmentierten Versionen 205 für das Training verwendet werden.

[0043] Das Training erfolgt damit beispielsweise selbstüberwacht mittels Kontrast-Verlust (engl. contrastive loss), siehe Referenz 1 für ein Beispiel.

[0044] Eine Augmentierung 204 kann als (geometrische) Abbildung jedes Pixels des Kamerabilds 203 auf eine entsprechende augmentierte Version 205 angesehen werden, d.h. jede Augmentierung bewirkt eine Positionsänderung von Pixelwerten. Ein Pixel im augmentierten Kamerabild entspricht einem anderen Pixel im ursprünglichen Kamerabild, wenn die Augmentierung den Pixelwert des anderen Pixels im ursprünglichen Kamerabild auf ihn abgebildet hat. Indem diese Abbildung (Positionsänderung) mitverfolgt wird, kann eine Korrespondenz zwischen Pixeln des Kamerabilds 203 und seiner Augmentierung 204 (oder auch zwischen Augmentierungen 204) hergestellt werden, also einander entsprechende Pixel (und dadurch auch Objektstellen) ermittelt werden. Damit können Positiv-Paare von Pixeln (die sich entsprechen) und Negativ-Paare von Pixeln (die sich nicht entsprechen) für das Kontrast-Lernen ermittelt werden. Beispielsweise werden für jedes gesampelte Positiv-Paar mehrere Negativ-Paare gesampelt. Pixel in zwei augmentierten Versionen des Kamerabilds entsprechen sich, wenn sie beide dem gleichen Pixel im ursprünglichen Kamerabild entsprechen. Die geometrischen Augmentierungen können auch durch nachgeschaltete weitere Augmentierungen (Farb- oder Helligkeitsänderungen, Rauschen etc.) ergänzt werden, um die Robustheit des maschinellen Lernmodells zu erhöhen.

[0045] Zusammen mit den (z.B. RGB-)Kamerabildern 203 (für unterschiedliche Objekte und/oder Szenen) können die Eingabedaten für das Trainingsverfahren auch zugehörige Objektmasken enthalten, die beispielsweise dadurch ermittelt werden können, dass ein Tiefenbild einer Szene mit einem Objekt mit dem Tiefenbild der Szene ohne das Objekt (also nur des Hintergrunds) verglichen wird.

[0046] Es kann dann (vor der Augmentierung 204) eine Vorverarbeitung vorgesehen sein, bei der, um die Datenmenge der Kamerabilder 203 zu reduzieren, die Kamerabilder auf eine kleinere Größe zugeschnitten werden, wobei jeweils die Objektmaske berücksichtigt wird, damit das Objekt in dem zugeschnittenen Kamerabild immer noch enthalten ist.

[0047] Die Augmentierungen sind z.B. Rotationen und perspektivische und affine Transformationen.

[0048] Die Ausgabe des maschinellen Lernmodells 201 für ein ihm zugeführtes Bild, d.h. ein (ggf. zugeschnittenes) Kamerabild 203 oder eine augmentierte Version 205 davon, ist ein Deskriptorbild mit (z.B.) den gleichen Dimensionen wie das zugeführte Bild aber mit einem Deskriptorvektor für jeden Pixel anstatt eines RGB-Werts. Das maschinelle Lernmodell 201 wird so trainiert, dass die Deskriptorvektoren von zwei Pixeln, die zum selben Ursprungspixel (vor der Augmentierung) gehören, d.h. die einander entsprechen, eng beieinander liegen. Die Deskriptorvektoren voneinander nicht entsprechenden Pixeln sollen hingegen im Deskriptorraum weit voneinander entfernt sein.

[0049] Fig. 3 zeigt zwei Trainingsdatenbilder 301, z.B. augmentierte Versionen 301 eines Kamerabilds. Eine Stelle des darin gezeigten Objekts 302 wird gesampelt und aus der Kenntnis über die Augmentierungen die korrespondierenden (entsprechende) Pixel in den beiden Trainingsdatenbildern 301 ermittelt. Beide Trainingsdatenbilder 301 werden durch das maschinelle Lernmodell auf jeweils ein Deskriptorbild 302 abgebildet (wobei hier sehr vereinfacht verschiedene Deskriptorwertbereiche mittels Schraffierungen dargestellt werden). Das maschinelle Lernmodell wird derart trainiert, dass die Deskriptorwerte der Pixel möglichst nah beieinander liegen und von den Deskriptorwerten anderer Pixel möglichst weit entfernt sind (im Mittel über viele Trainingsbilder und gesampelte Stellen). Die Korrespondenz von Pixeln in den Trainingsdatenbildern 201 und den Deskriptorbildern 302 ist durch die Pfeile angedeutet.

[0050] Eines der Trainingsdatenbilder 301 kann auch das ursprüngliche Kamerabild selbst sein.

[0051] Zusammengefasst wird gemäß verschiedenen Ausführungsformen ein Verfahren bereitgestellt, wie in Fig. 4 dargestellt.

[0052] Fig. 4 zeigt ein Ablaufdiagramm 400, das ein Verfahren zum Trainieren eines maschinellen Lernmodells zum Erzeugen von Deskriptorbildern zu Bildern von ein oder mehreren Objekten gemäß einer Ausführungsform darstellt.

[0053] In 401 werden mehrere Kamerabilder aufgenommen, wobei jedes Kamerabild ein oder mehrere Objekte zeigt.

[0054] In 402 werden für jedes Kamerabild

- in 403 ein oder mehrere augmentierte Versionen des Kamerabilds durch Anwenden, für jede augmentierte Version des Kamerabilds, einer jeweiligen Augmentierung auf das Kamerabild, wobei die Augmentierung eine Positionsänderung von Pixelwerten des Kamerabilds aufweist, erzeugt;

- in 404 ein oder mehrere Trainingsbildpaaren erzeugt, die jeweils das Kamerabild und eine Augmentierung des Kamerabilds oder zwei augmentierte Versionen des Kamerabilds aufweisen;

- in 405 für jedes Trainingsbildpaar gemäß der Positionsänderung von Pixelwerten, die die Augmentierung aufweist, mit der die augmentierte Version erzeugt wurde, falls das Trainingsbildpaar eine augmentierte Version des Kamerabilds aufweist, oder gemäß den Positionsänderungen von Pixelwerten, die die beiden Augmentierungen aufweisen, mit der die augmentierten Versionen erzeugt wurden, falls das Trainingsbildpaar zwei augmentierte Versionen des Kamerabilds aufweist, einander entsprechende Pixel und einander nicht entsprechende Pixel des Trainingsbildpaares ermittelt.

[0055] In 406 wird das maschinelle Lernmodells mittels Kontrast-Verlust unter Verwendung der Trainingsbildpaare trainiert. Dabei werden Deskriptorwerte, die das maschinelle Lernmodell füreinander entsprechende Pixel erzeugt, als Positiv-Paare und Deskriptorwerte, die das maschinelle Lernmodell füreinander nicht entsprechende Pixel erzeugt, als Negativ-Paare verwendet.

[0056] Es sollte beachtet werden, dass die augmentierten Versionen nicht notwendig erst erstellt werden, nachdem alle Kamerabilder aufgenommen wurden. Dies kann beispielsweise auch im Wechsel geschehen. Die Reihenfolge ist also nicht auf die oben genannte Reihenfolge eingeschränkt.

[0057] Das Verfahren von **Fig. 4** kann durch einen oder mehrere Computer mit einer oder mehreren Datenverarbeitungseinheiten durchgeführt werden. Der Begriff „Datenverarbeitungseinheit“ kann als irgendein Typ von Entität verstanden werden, der die Verarbeitung von Daten oder Signalen ermöglicht. Die Daten oder Signale können beispielsweise gemäß mindestens einer (d.h. einer oder mehr als einer) speziellen Funktion behandelt werden, die durch die Datenverarbeitungseinheit durchgeführt wird. Eine Datenverarbeitungseinheit kann eine analoge Schaltung, eine digitale Schaltung, eine Logikschaltung, einen Mikroprozessor, einen Mikrocontroller, eine Zentraleinheit (CPU), eine Graphikverarbeitungseinheit (GPU), einen Digitalsignalprozessor (DSP), eine integrierte Schaltung einer programmierbaren Gatteranordnung (FPGA) oder irgendeine Kombination davon umfassen oder aus dieser ausgebildet sein. Irgendeine andere Weise zum Implementieren der jeweiligen Funktionen, die hierin genauer beschrieben werden, kann auch als Datenverarbeitungseinheit oder Logikschaltungsanordnung verstanden werden. Es können ein oder mehrere der im Einzelnen hier beschriebenen Ver-

fahrensschritte durch eine Datenverarbeitungseinheit durch eine oder mehrere spezielle Funktionen ausgeführt (z. B. implementiert) werden, die durch die Datenverarbeitungseinheit durchgeführt werden.

[0058] Mittels des trainierten maschinellen Lernmodells kann (z.B. durch Verwenden des trainierten maschinellen Lernmodells zum Ermitteln einer Objektpose oder Ermittlung zu bearbeitender Stellen) letztendlich ein Steuersignal für eine Roboteranordnung erzeugt werden. Es können dazu relevante Stellen jeglicher Art von Objekten ermittelt werden, für die das maschinelle Lernmodell trainiert wurde. Der Begriff „Roboteranordnung“ kann als sich auf irgendein physikalisches System beziehend verstanden werden, wie z.B. eine computergesteuerte Maschine, ein Fahrzeug, ein Haushaltsgerät, ein Elektrowerkzeug, eine Fertigungsmaschine, einen persönlichen Assistenten oder ein Zugangssystem. Es wird eine Steuerungsvorschrift für das physikalische System gelernt und das physikalische System dann entsprechend gesteuert.

[0059] Beispielsweise werden Bilder mittels einer RGB-D(Farbbild plus Tiefe)-Kamera aufgenommen, durch das trainierte maschinelle Lernmodell (z.B. neuronale Netz) verarbeitet und relevanten Stellen im Arbeitsbereich der Roboteranordnung ermittelt, wobei die Roboteranordnung abhängig von den ermittelten Stellen gesteuert wird. Beispielsweise kann ein Objekt (d.h. seine Position und/oder Pose) in Eingangs-Sensordaten verfolgt werden.

[0060] Die Kamerabilder sind beispielsweise RGB-Bilder oder RGB-D(Farbbild plus Tiefe)-Bilder, können aber auch andere Arten von Kamerabildern wie (nur) Tiefenbilder oder Wärme-, Video-, Radar-, LiDAR-, Ultraschall-, oder Bewegungsbilder sein. Tiefenbilder sind nicht zwingend erforderlich. Die Ausgabe des trainierten maschinellen Lernmodells kann dazu verwendet werden, Objektposen zu ermitteln, beispielsweise zur Steuerung eines Roboters, z.B. für den Zusammenbau eines größeren Objekts aus Teilobjekten, das Bewegen von Objekten etc. Die Herangehensweise von **Fig. 4** kann für jegliches Posenermittlungsverfahren verwendet werden. Aus relevanten Stellen, die aus den Deskriptorbildern ermittelt werden, kann z.B. die Lage von Stellen eines Objekts ermittelt werden, an denen das Objekt von einem Roboter gegriffen (oder auch angesaugt) werden kann, und solchen, an denen es nicht gegriffen werden sollte. Sie liefern somit beispielsweise wertvolle Informationen für eine Manipulationsaufgabe.

[0061] Obwohl spezielle Ausführungsformen hier dargestellt und beschrieben wurden, wird vom Fachmann auf dem Gebiet erkannt, dass die speziellen Ausführungsformen, die gezeigt und beschrieben sind, gegen eine Vielfalt von alternativen und/oder

äquivalenten Implementierungen ausgetauscht werden können, ohne vom Schutzbereich der vorliegenden Erfindung abzuweichen. Diese Anmeldung soll irgendwelche Anpassungen oder Variationen der speziellen Ausführungsformen abdecken, die hier erörtert sind. Daher ist beabsichtigt, dass diese Erfindung nur durch die Ansprüche und die Äquivalente davon begrenzt ist.

Patentansprüche

1. Verfahren zum Trainieren eines maschinellen Lernmodells zum Erzeugen von Deskriptorbildern zu Bildern von ein oder mehreren Objekten aufweisend:

Aufnehmen von mehreren Kamerabildern, wobei jedes Kamerabild ein oder mehrere Objekte zeigt;

Für jedes Kamerabild

Erzeugen von ein oder mehreren augmentierten Versionen des Kamerabilds durch Anwenden, für jede augmentierte Version des Kamerabilds, einer jeweiligen Augmentierung auf das Kamerabild, wobei die Augmentierung eine Positionsänderung von Pixelwerten des Kamerabilds aufweist;

Erzeugen von ein oder mehreren Trainingsbildpaaren, die jeweils das Kamerabild und eine Augmentierung des Kamerabilds oder zwei augmentierte Versionen des Kamerabilds aufweisen; und

Ermitteln, für jedes Trainingsbildpaar, gemäß der Positionsänderung von Pixelwerten, die die Augmentierung aufweist, mit der die augmentierte Version erzeugt wurde, falls das Trainingsbildpaar eine augmentierte Version des Kamerabilds aufweist, oder gemäß den Positionsänderungen von Pixelwerten, die die beiden Augmentierungen aufweisen, mit der die augmentierten Versionen erzeugt wurden, falls das Trainingsbildpaar zwei augmentierte Versionen des Kamerabilds aufweist, voneinander entsprechenden Pixeln und einander nicht entsprechenden Pixeln des Trainingsbildpaares und Trainieren des maschinellen Lernmodells mittels Kontrast-Verlust unter Verwendung der Trainingsbildpaare wobei Deskriptorwerte, die das maschinelle Lernmodell füreinander entsprechende Pixel erzeugt, als Positiv-Paare und Deskriptorwerte, die das maschinelle Lernmodell füreinander nicht entsprechende Pixel erzeugt, als Negativ-Paare verwendet werden.

2. Verfahren nach Anspruch 1, wobei die Augmentierung eine Rotation, eine perspektivische Transformation und/oder eine affine Transformation aufweist.

3. Verfahren nach Anspruch 1 oder 2, aufweisend, vor dem Erzeugen der augmentierten Versionen und der Trainingsbildpaare, Zuschneiden der Kamerabilder unter Berücksichtigung von Objektmasken der ein oder mehreren Objekten.

4. Verfahren nach einem der Ansprüche 1 bis 3, wobei das maschinelle Lernmodell ein neuronales Netz ist.

5. Verfahren nach einem der Ansprüche 1 bis 4, wobei die mehreren Kamerabilder aus derselben Perspektive aufgenommen werden.

6. Verfahren zum Steuern eines Roboters zum Aufnehmen oder Bearbeiten eines Objekts, aufweisend:

Trainieren eines maschinellen Lernmodells nach einem der Ansprüche 1 bis 5;

Aufnehmen eines Kamerabilds, das das Objekt in einem aktuellen Steuerszenario zeigt;

Zuführen des Kamerabilds zu dem maschinellen Lernmodell zum Erzeugen eines Deskriptorbilds;

Ermitteln der Position einer Stelle zum Aufnehmen oder Bearbeiten des Objekts in dem aktuellen Steuerszenario aus dem Deskriptorbild; und

Steuern des Roboters entsprechend der ermittelten Position.

7. Verfahren nach Anspruch 6, aufweisend:

Identifizieren einer Referenzstelle in einem Referenzbild;

Ermitteln eines Deskriptors der identifizierten Referenzstelle durch Zuführen des Referenzbilds zu dem maschinellen Lernmodell;

Ermitteln der Position der Referenzstelle in dem aktuellen Steuerszenario durch Suchen des ermittelten Deskriptors in dem aus dem Kamerabild erzeugten Deskriptorbild; und

Ermitteln der Position der Stelle zum Aufnehmen oder Bearbeiten des Objekts in dem aktuellen Steuerszenario aus der ermittelten Position der Referenzstelle.

8. Steuereinrichtung, die eingerichtet ist, ein Verfahren nach einem der Ansprüche 1 bis 7 durchzuführen.

9. Computerprogramm mit Befehlen, die, wenn sie durch einen Prozessor ausgeführt werden, bewirken, dass der Prozessor ein Verfahren nach einem der Ansprüche 1 bis 7 durchführt.

10. Computerlesbares Medium, das Befehle speichert, die, wenn sie durch einen Prozessor ausgeführt werden, bewirken, dass der Prozessor ein Verfahren nach einem der Ansprüche 1 bis 7 durchführt.

Es folgen 3 Seiten Zeichnungen

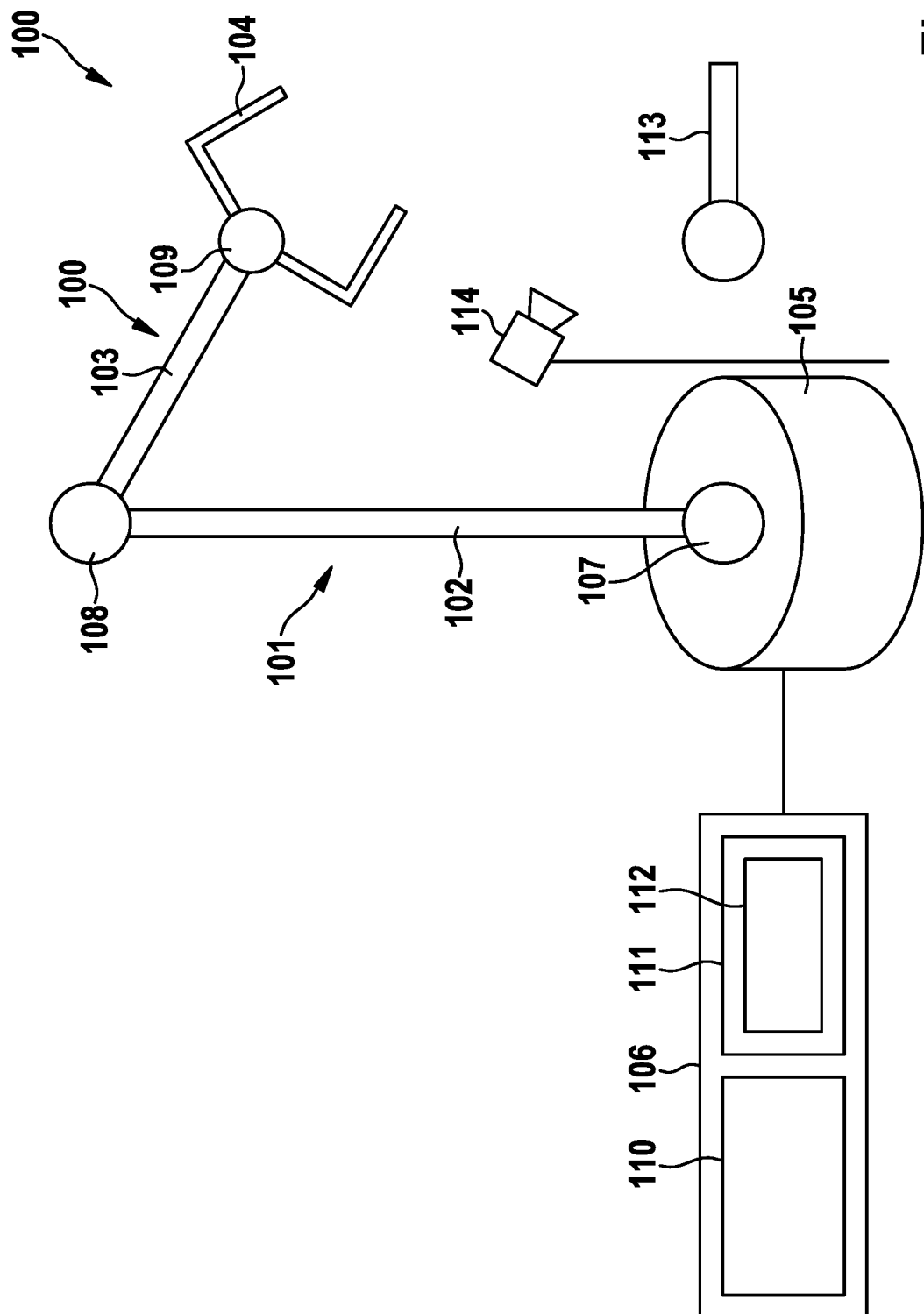


Fig. 1

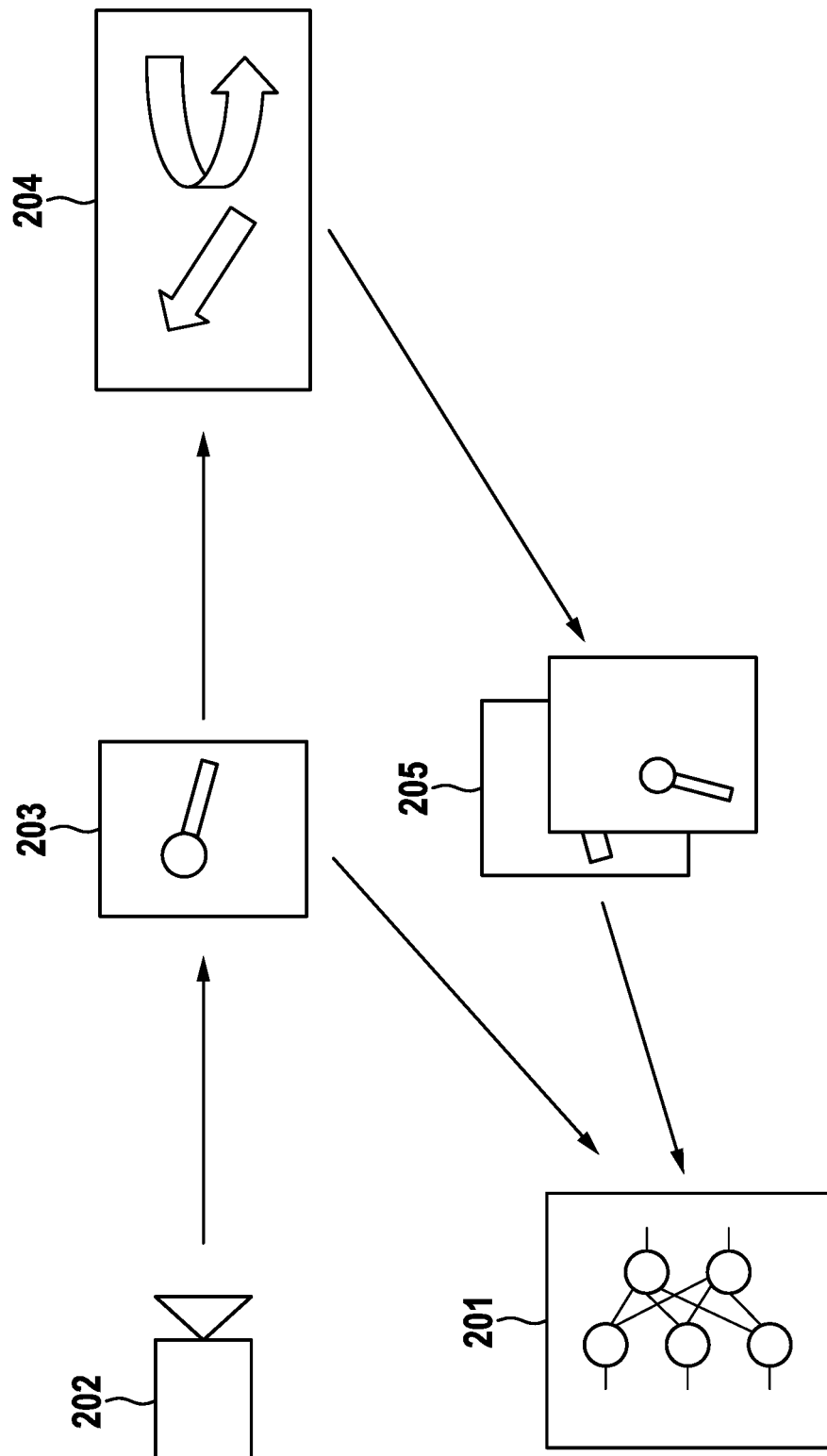


Fig. 2

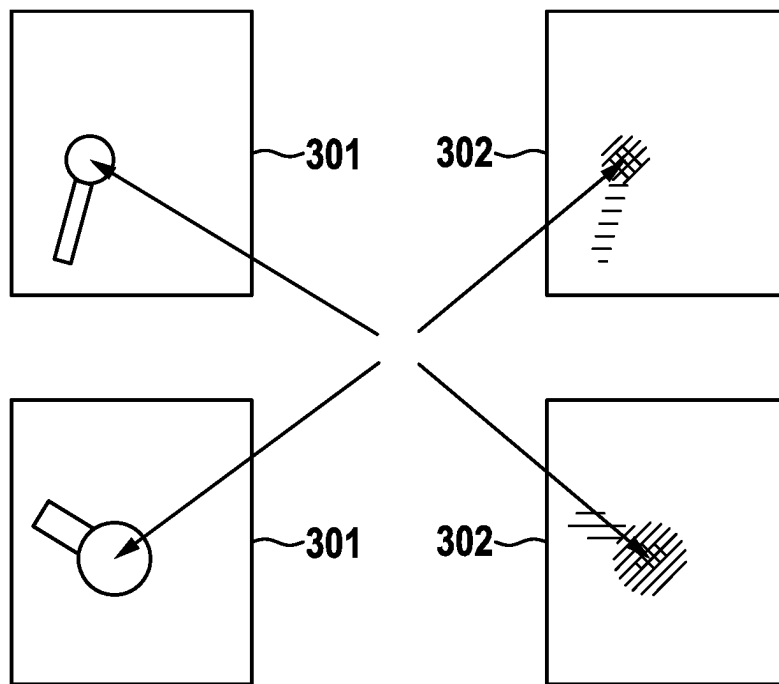


Fig. 3

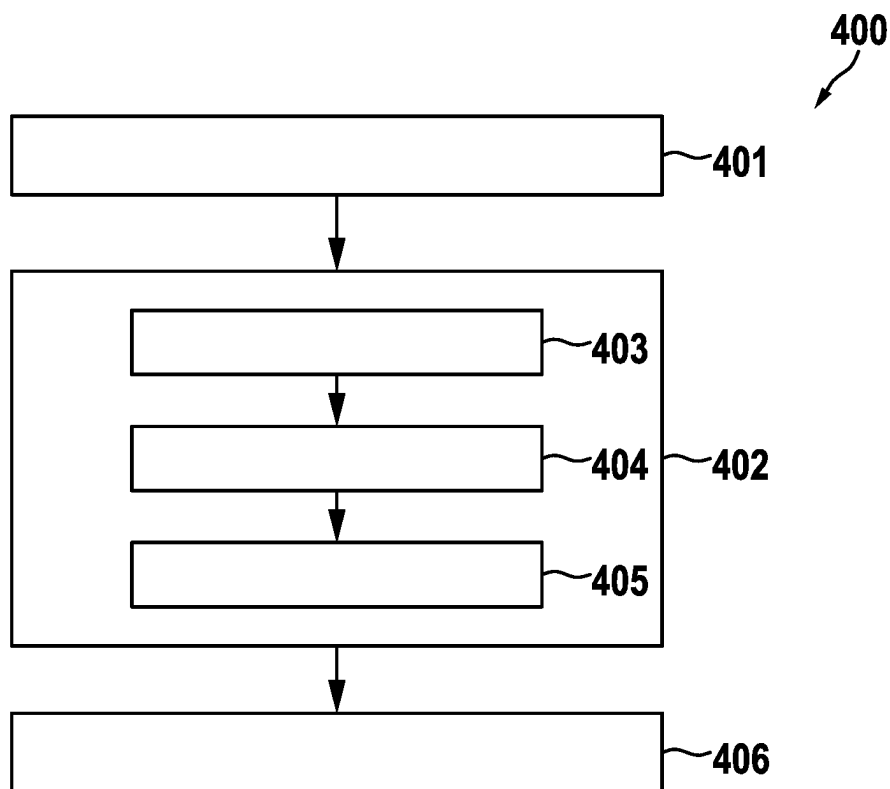


Fig. 4