



(10) **DE 10 2019 003 621 A1** 2020.01.02

(12)

Offenlegungsschrift

(21) Aktenzeichen: **10 2019 003 621.2**

(22) Anmeldetag: **23.05.2019**

(43) Offenlegungstag: **02.01.2020**

(51) Int Cl.: **G01S 7/48 (2006.01)**

G01S 17/93 (2006.01)

(71) Anmelder:
Daimler AG, 70327 Stuttgart, DE

(72) Erfinder:
**Triess, Larissa, 70565 Stuttgart, DE; Peter, David,
Dr.rer.nat., 70565 Stuttgart, DE**

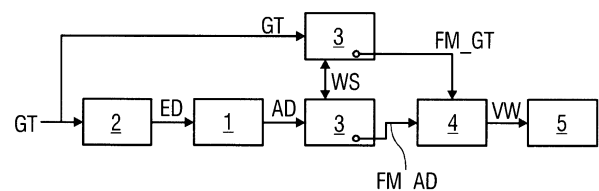
Mit Einverständnis des Anmelders offengelegte Anmeldung gemäß § 31 Abs. 2 Ziffer 1 PatG

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen.

(54) Bezeichnung: **Verfahren zur Verarbeitung von Lidarsensordaten**

(57) Zusammenfassung: Die Erfindung betrifft ein Verfahren zur Verarbeitung von Lidarsensordaten.

Erfindungsgemäß wird ein faltendes neuronales Netzwerk (1) verwendet, mittels welchem anhand der von einem Lidarsensor, insbesondere eines Fahrzeugs, erfassten niedrig aufgelösten Lidarsensordaten höher aufgelöste Entfernung-Punktwolken generiert werden.



Beschreibung

[0001] Die Erfindung betrifft ein Verfahren zur Verarbeitung von Lidarsensordaten.

[0002] Aus dem Stand der Technik ist eine Verarbeitung von Lidarsensordaten in Fahrzeugen allgemein bekannt.

[0003] In „Perceptual Losses for Real-Time Style Transfer and Super-Resolution“, Justin Johnson, Alexandre Alahi, Li Fei-Fei, Department of Computer Science, Stanford University, werden Bildtransformationsprobleme behandelt, bei denen ein Eingangsbild in ein Ausgangsbild transformiert wird. Es werden Wahrnehmungsverlustfunktionen für das Training von Feed-Forward-Netzwerken für Bildtransformationsaufgaben verwendet.

[0004] Der Erfindung liegt die Aufgabe zu Grunde, ein gegenüber dem Stand der Technik verbessertes Verfahren zur Verarbeitung von Lidarsensordaten anzugeben.

[0005] Die Aufgabe wird erfindungsgemäß gelöst durch ein Verfahren zur Verarbeitung von Lidarsensordaten mit den Merkmalen des Anspruchs 1.

[0006] Vorteilhafte Ausgestaltungen der Erfindung sind Gegenstand der Unteransprüche.

[0007] In einem erfindungsgemäßen Verfahren zur Verarbeitung von Lidarsensordaten, insbesondere zum künstlichen Erzeugen von hoch auflösenden Lidar-Scans mittels eines niedrig auflösenden Lidarsensors, wird ein faltendes neuronales Netzwerk (Convolutional Neural Network - CNN) verwendet, mittels welchem anhand der vom Lidarsensor, insbesondere eines Fahrzeugs, erfassten niedrig aufgelösten Lidarsensordaten, d. h. anhand von niedrig aufgelösten Lidar-Scans, höher aufgelöste, insbesondere hochauflöste, insbesondere realistische, Entfernungspunktwolken generiert werden.

[0008] Vorteilhafterweise wird/werden bei einem Trainieren des faltenden neuronalen Netzwerks mindestens eine Wahrnehmungsverlustfunktion oder Wahrnehmungsverlustfunktionen angewandt. Durch dieses Anwenden einer oder mehrerer solcher Wahrnehmungsverlustfunktionen (perceptual loss functions) werden lückenhafte Entfernungswerte ausgeglichen. Diese Wahrnehmungsverlustfunktion ist dabei insbesondere nicht eine normale Berechnung, sondern sie ist insbesondere in sich selbst auch ein tiefes neuronales Netzwerk.

[0009] Vorteilhafterweise werden zum Trainieren des faltenden neuronalen Netzwerks mittels eines Trainingsdatengenerators aus Ground-Truth-Daten die niedrig aufgelösten Lidarsensordaten generiert,

insbesondere auf eine niedrigere Auflösung heruntergerechnet, und dem faltenden neuronalen Netzwerk als Eingangsdaten zugeführt, welches daraus die höher aufgelösten Entfernungspunktwolken als Ausgangsdaten generiert. D. h. das faltende neuronale Netzwerk wird trainiert, aus diesen heruntergerechneten niedrig aufgelösten Eingangsdaten hoch aufgelöste Ausgangsdaten zu generieren.

[0010] Das Herzstück des Verfahrens ist somit das faltende neuronale Netzwerk, das direkt mit dem Lidar-Scan, d. h. mit den Lidarsensordaten, arbeitet. Es verwendet Lidar-Entfernungsmessungen mit niedriger Auflösung als Eingangsdaten und generiert realistische Lidar-Scans mit hoher Auflösung als Ausgangsdaten. Das faltende neuronale Netzwerk wird aus Sequenzen von Lidar-Aufzeichnungen ohne manuelle Anmerkungen trainiert. Das Verfahren berücksichtigt insbesondere fehlende Entfernungsmessungen, die bei Lidar-Punktwolken stets möglich sind. Um realistische Ausgangsdaten zu erzielen, wird die Wahrnehmungsverlustfunktion zum Trainieren des faltenden neuronalen Netzwerks verwendet. Diese Wahrnehmungsverlustfunktion verwendet ein separat trainiertes Merkmalsextraktionsnetzwerk (für eine semantische Kennzeichnung), um die semantische Bedeutung und das visuelle Erscheinungsbild der Punktwolke auf hohem Niveau zu erhalten.

[0011] Lidarsensoren, auch als Lidar-Scanner bezeichnet, sind unverzichtbare Sensoren für fortschrittliche Fahrerassistenzsysteme von Fahrzeugen und für selbstfahrende Fahrzeuge. Heutige Lidarsensoren verfügen über eine sehr begrenzte (vertikale) Auflösung, die aus Sicherheitssicht nicht ausreichend ist. Die Entwicklung in Richtung Sensoren mit höheren Auflösungen schreitet jedoch stetig voran. Dies bedeutet, dass zukünftige Fahrzeuge mit Lidarsensoren ausgestattet werden, die derzeit noch nicht verfügbar sind. Dies ist ein Problem für Softwareentwickler, da sie die erforderliche Nachverarbeitungssoftware (Umgebungswahrnehmungsalgorithmen) schreiben müssen, ohne die richtigen Eingangsdaten zu haben. Das erfindungsgemäße Verfahren löst dieses Problem, denn es ermöglicht ein Synthetisieren realistischer Lidar-Scans mit hoher Auflösung aus Eingangsdaten, d. h. Lidarsensordaten, mit niedriger Auflösung.

[0012] Die erfindungsgemäße Lösung ermöglicht somit Entwicklern die Verwendung von Lidarsensoren mit niedriger Auflösung bei der Verarbeitung zukünftiger hochauflöster Lidar-Scans, ohne auf die eigentliche Hardware warten zu müssen.

[0013] Zudem kann das Verfahren auch bereits in derzeit verfügbaren Fahrzeugen und somit auch bei den hier vorhandenen niedrig auflösenden Lidarsensoren verwendet werden, um aus den erfassten niedrig aufgelösten Lidarsensordaten die hochauflöste

ten realistischen Entfernungs-Punktwolken zu generieren und dadurch einen Sicherheitsgewinn für die Fahrerassistenzsysteme zu erzielen.

[0014] Ausführungsbeispiele der Erfindung werden im Folgenden anhand von Zeichnungen näher erläutert.

[0015] Dabei zeigen:

Fig. 1 eine schematische Darstellung eines Trainieren eines faltenden neuronalen Netzwerks für ein Verfahren zur Verarbeitung von Lidarsensordaten, und

Fig. 2 schematisch eine Architektur des faltenden neuronalen Netzwerks.

[0016] Einander entsprechende Teile sind in allen Figuren mit den gleichen Bezugszeichen versehen.

[0017] Im Folgenden wird anhand der **Fig. 1** und **Fig. 2** ein Verfahren zur Verarbeitung von Lidarsensordaten, insbesondere zum künstlichen Erzeugen von hoch auflösenden Lidar-Scans mittels eines niedrig auflösenden Lidarsensors, beschrieben. Der Lidarsensor ist insbesondere ein Lidarsensor eines Fahrzeugs oder für ein Fahrzeug. Derartige Lidarsensoren, auch als Lidar-Scanner bezeichnet, sind unverzichtbare Sensoren für fortschrittliche Fahrerassistenzsysteme von Fahrzeugen und für selbstfahrende Fahrzeuge.

[0018] Heutige Lidarsensoren verfügen über eine sehr begrenzte (vertikale) Auflösung, die aus Sicherheitssicht nicht ausreichend ist. Die Entwicklung in Richtung Sensoren mit höheren Auflösungen schreitet jedoch stetig voran. Dies bedeutet, dass zukünftige Fahrzeuge mit Lidarsensoren ausgestattet werden, die derzeit noch nicht verfügbar sind. Dies ist ein Problem für Softwareentwickler, da sie die erforderliche Nachverarbeitungssoftware (Umgebungswahrnehmungsalgorithmen) schreiben müssen, ohne die richtigen Eingangsdaten zu haben.

[0019] In dem hier beschriebenen Verfahren ist daher vorgesehen, dass mittels eines faltenden neuronalen Netzwerks **1** (Convolutional Neural Network - CNN) anhand der vom Lidarsensor erfassten niedrig aufgelösten Lidarsensordaten, d. h. anhand von niedrig aufgelösten Lidar-Scans, hochaufgelöste realistische Entfernungs-Punktwolken generiert werden. Um dies zu ermöglichen, wird das faltende neuronale Netzwerk **1** zunächst trainiert, wie in **Fig. 1** schematisch dargestellt.

[0020] Zum Trainieren des faltenden neuronalen Netzwerks **1** werden mittels eines Trainingsdatengenerators **2** aus hoch aufgelösten Ground-Truth-Daten **GT** die niedrig aufgelösten Lidarsensordaten generiert, insbesondere auf eine niedrigere Auflösung her-

untergerechnet, und dem faltenden neuronalen Netzwerk **1** als niedrig aufgelöste Eingangsdaten **ED** zugeführt, welches daraus die hochaufgelösten Entfernungs-Punktwolken als hochaufgelöste Ausgangsdaten **AD** generiert. D. h. das faltende neuronale Netzwerk **1** wird trainiert, aus diesen heruntergerechneten niedrig aufgelösten Eingangsdaten **ED** hoch aufgelöste Ausgangsdaten **AD** zu generieren.

[0021] Bei diesem Trainieren des faltenden neuronalen Netzwerks **1** wird vorteilhafterweise eine Wahrnehmungsverlustfunktion (perceptual loss function) angewandt. Dadurch werden lückenhafte Entfernungswerte ausgeglichen.

[0022] Im Beispiel gemäß **Fig. 1** ist zudem ein Merkmalsextraktionsnetzwerk **3**, insbesondere bestehend aus einem LiLaNet (Feature Extractor), vorgesehen. Ein solches LiLaNet wird beispielsweise in <https://arxiv.org/pdf/1804.09915.pdf> beschrieben. Für das hier beschriebene Verfahren wurde die Architektur in dem Sinne verändert, dass eine Anzahl von Filtern innerhalb dieses Netzwerks auf 22% der (oben angegebenen) Originalveröffentlichung reduziert wurde, um das Netzwerk trainierbar zu machen für den hier beschriebenen Zweck.

[0023] Dieses LiLaNet ist ein Punktwolken-Merkmalsextraktionsnetzwerk. Ursprünglich, d. h. in der oben angegebenen Originalveröffentlichung, wird das LiLaNet für eine semantische Segmentierung von Punktwolken verwendet. Für das hier beschriebene Verfahren wird es jedoch als Feature Extractor (zur Merkmalsextraktion) verwendet, da es sich hierfür gut eignet. Es wird hier verwendet, um einen sogenannten Wahrnehmungsverlust (perceptual loss) zu ermitteln, der zwei Merkmalskarten **FM_GT**, **FM_AD** (Feature Maps) in einem punktwweisen L1-Regressionsverlust vergleicht. L1 ist dabei eine Regressionsverlustfunktion auf Basis der geringsten absoluten Abweichungen (least absolute deviations). Dabei wird die eine Merkmalskarte **FM_GT** vom Merkmalsextraktionsnetzwerk **3**, insbesondere vom LiLaNet, aus den Ground-Truth-Daten **GT** extrahiert und die andere Merkmalskarte **FM_AD** wird vom Merkmalsextraktionsnetzwerk **3**, insbesondere vom LiLaNet, aus den hoch aufgelösten Ausgangsdaten **AD** des faltenden neuronalen Netzwerks **1** extrahiert bzw. generiert.

[0024] Die in **Fig. 1** schematisch dargestellten beiden Blöcke des Merkmalsextraktionsnetzwerks **3** sind dabei exakt das gleiche Netzwerk, welches auch nicht mehr trainiert wird (vortrainiert). Es ist in **Fig. 1** nur als zwei Blöcke dargestellt, damit klar ersichtlich ist, dass die Ground-Truth-Daten **GT** und die Ausgangsdaten **AD** nicht zusammen in den gleichen Block gehen, sondern voneinander getrennt. Gewichte (weights) zwischen diesen beiden Blöcken sind die

gleichen (shared weights), in **Fig. 1** dargestellt als weight sharing **WS**.

[0025] Die Merkmalskarten **FM_GT**, **FM_AD** werden dann einer punktweisen Regressionsverlustkomponente **4** (Loss) zugeführt, um den Wahrnehmungsverlust zu ermitteln. Im dargestellten Beispiel wird ein von der Regressionsverlustkomponente **4** ermittelter Verlustwert **VW** dann einem Optimierer **5**, insbesondere einem Adam Optimizer, zugeführt.

[0026] **Fig. 2** zeigt beispielhaft eine Architektur des faltenden neuronalen Netzwerks **1**, inspiriert durch und insbesondere abgeleitet aus J. Johnson, A. Alahi, and L. Fei-Fei, „Perceptual losses for real-time style transfer and super-resolution,“ in ECCV, 2016. Die Architektur umfasst einen Faltungsblock **FB1** mit 64 Filtern und einer Kerngröße von 9x9, vier Restwertblöcke **RB** (Residual Blocks), einen transponierten Faltungsblock **TFB** mit 64 Filtern und einer Kerngröße von 4x1, in welchem das tatsächliche Hochrechnen auf die hohe Auflösung erfolgt, und einen Faltungsblock **FB2** mit einem Filter und einer Kerngröße von 9x9. Jeder Restwertblock **RB** umfasst zwei Faltungsblöcke **RFB** mit 64 Filtern und einer Kerngröße von 3x3, gefolgt jeweils von einem Stapelnormalisierungsblock **SB** (Batch Normalization). Zwischen den beiden Restwertblöcken **RB** ist eine Aktivierungsfunktion **AF** angeordnet, genauer gesagt eine ReLU (Rectifier Linear Unit) activation function.

[0027] Zum Verständnis des hier beschriebenen Verfahrens ist die obige detaillierte Beschreibung der in **Fig. 2** dargestellten Architektur nicht unbedingt erforderlich. Die für das Verfahren insbesondere wichtige Makroarchitektur, welche die Neuerung gegenüber dem Stand der Technik in der Verarbeitung der Lidardaten darstellt, ist bereits in **Fig. 1** gezeigt. Die Neuerung besteht insbesondere auch darin, dass anstatt eines klassischen Regressions-Loss (direkt nach dem ersten Netzwerk, hier nach dem faltenden neuronalen Netzwerk **1**, zwischen den Ausgangsdaten **AD** und den Ground-Truth-Daten **GT**), ein zweites Netzwerk, hier in Form des aus dem LiLaNet bestehenden Merkmalsextraktionsnetzwerks **3**, verwendet wird, um den eigentlich Loss, d. h. insbesondere den Wahrnehmungsverlust, zwischen der vom Merkmalsextraktionsnetzwerk **3**, insbesondere vom LiLaNet, aus den hoch aufgelösten Ausgangsdaten **AD** des faltenden neuronalen Netzwerks **1** extrahierten bzw. generierten Merkmalskarte **FM_AD** und der vom Merkmalsextraktionsnetzwerk **3**, insbesondere vom LiLaNet, aus den Ground-Truth-Daten **GT** extrahierten Merkmalskarte **FM_GT** in der punktweisen Regressionsverlustkomponente **4** zu berechnen. Nur so kann eine realistische Erzeugung der Punktwolken erfolgen.

[0028] Das Herzstück des Verfahrens ist somit das faltende neuronale Netzwerk **1**, das direkt mit dem

Lidar-Scan, d. h. mit den Lidarsensordaten, arbeitet. Es verwendet Lidar-Entfernungsmessungen mit niedriger Auflösung als Eingangsdaten **ED** und generiert realistische Lidar-Scans mit hoher Auflösung als Ausgangsdaten **AD**. Das faltende neuronale Netzwerk **1** wird aus Sequenzen von Lidar-Aufzeichnungen ohne manuelle Anmerkungen trainiert. Das Verfahren berücksichtigt insbesondere fehlende Entfernungsmessungen, die bei Lidar-Punktwolken stets möglich sind. Um realistische Ausgangsdaten **AD** zu erzielen, wird die Wahrnehmungsverlustfunktion zum Trainieren des faltenden neuronalen Netzwerks **1** verwendet. Diese Wahrnehmungsverlustfunktion verwendet ein separat trainiertes Merkmalsextraktionsnetzwerk (für eine semantische Kennzeichnung), um die semantische Bedeutung und das visuelle Erscheinungsbild der Punktwolke auf hohem Niveau zu erhalten.

[0029] Das beschriebene Verfahren ermöglicht somit ein Synthetisieren realistischer Lidar-Scans mit hoher Auflösung aus Eingangsdaten **ED**, d. h. Lidarsensordaten, mit niedriger Auflösung.

Bezugszeichenliste

1	faltendes neuronales Netzwerk
2	Trainingsdatengenerator
3	Merkmalsextraktionsnetzwerk
4	Regressionsverlustkomponente
5	Optimierer
AD	Ausgangsdaten
AF	Aktivierungsfunktion
ED	Eingangsdaten
FM_AD, FM_GT	Merkmalskarte
FB1, FB2	Faltungsblock
RFB	Faltungsblock
GT	Ground-Truth-Daten
RB	Restwertblock
SB	Stapelnormalisierungsblock
TFB	transponierter Faltungsblock
VW	Verlustwert
WS	weight sharing

Patentansprüche

1. Verfahren zur Verarbeitung von Lidarsensordaten, **dadurch gekennzeichnet**, dass ein faltendes neuronales Netzwerk (1) verwendet wird, mittels welchem anhand der von einem Lidarsensor, insbesondere eines Fahrzeugs, erfassten niedrig aufgelösten Lidarsensordaten höher aufgelöste Entfernungspunktwolken generiert werden.

2. Verfahren nach Anspruch 1, **dadurch gekennzeichnet**, dass bei einem Trainieren des faltenden neuronalen Netzwerks (1) mindestens eine Wahrnehmungsverlustfunktion angewandt wird, um lückenhafte Entfernungswerte auszugleichen.

3. Verfahren nach einem der vorhergehenden Ansprüche, **dadurch gekennzeichnet**, dass zum Trainieren des faltenden neuronalen Netzwerks (1) mittels eines Trainingsdatengenerators (2) aus Ground-Truth-Daten (GT) die niedrig aufgelösten Lidarsensordaten generiert werden und dem faltenden neuronalen Netzwerk (1) zugeführt werden, welches daraus die höher aufgelösten Entfernungspunktwolken generiert.

Es folgt eine Seite Zeichnungen

Anhängende Zeichnungen

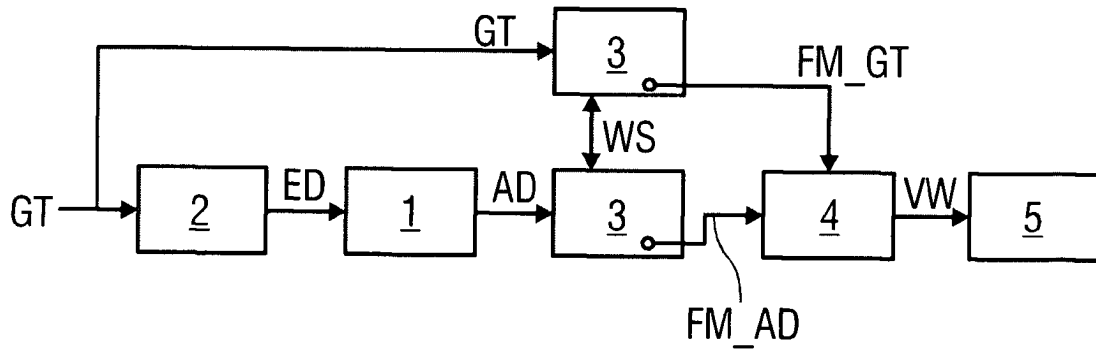


FIG 1

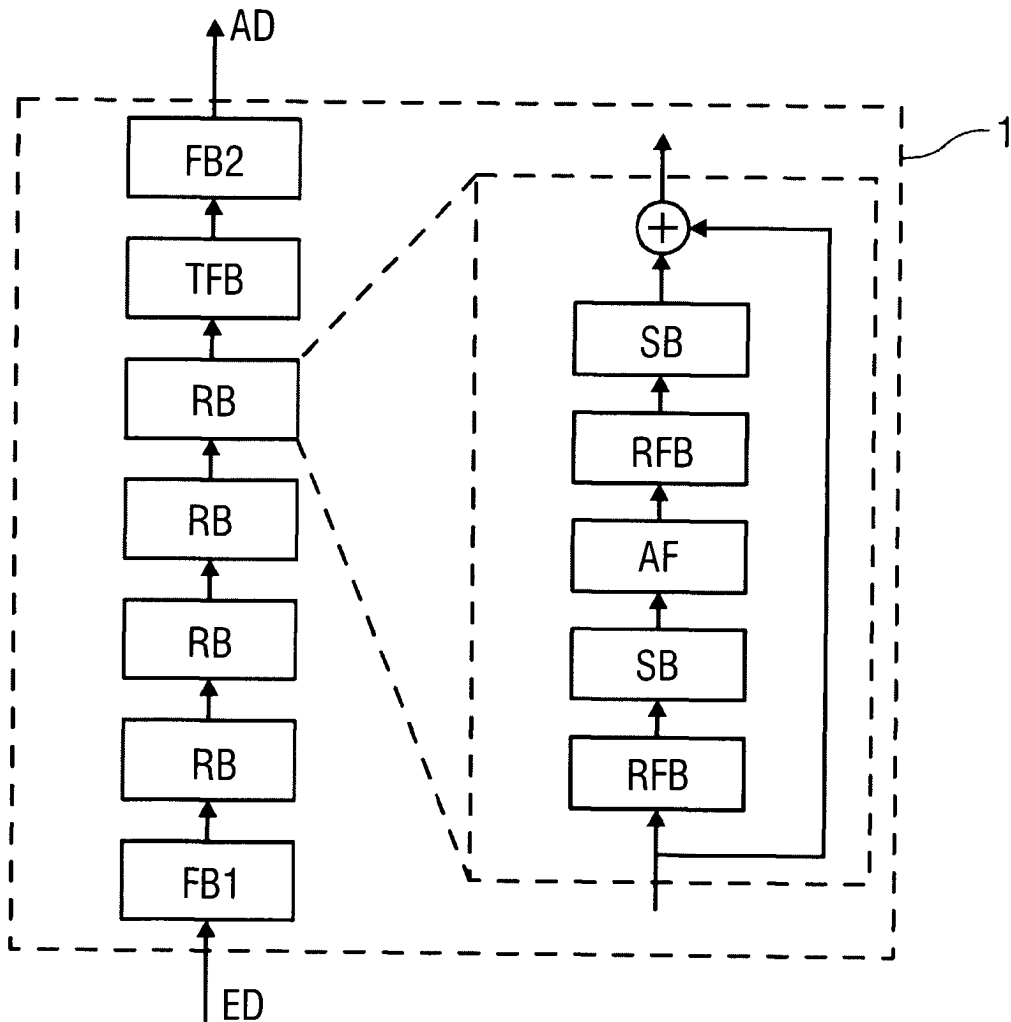


FIG 2