



(10) **DE 10 2018 201 290 A1** 2019.08.01

(12)

Offenlegungsschrift

(21) Aktenzeichen: **10 2018 201 290.3**

(22) Anmeldetag: **29.01.2018**

(43) Offenlegungstag: **01.08.2019**

(51) Int Cl.: **G06N 3/08** (2006.01)

(71) Anmelder:

OSRAM GmbH, 80807 München, DE

(72) Erfinder:

**Belagiannis, Vasileios, 81245 München, DE;
Farshad, Azade, 80939 München, DE; Galasso,
Fabio, 85748 Garching, DE**

(56) Ermittelter Stand der Technik:

BPatG 17 W (pat) 6/06, 09.03.2010

**Ba, Jimmy; Caruana, Rich: Do Deep Nets
Really Need to be Deep?. In: Advances in Neural
Information Processing Systems 27, 2014, S.
2654-2662. [http://papers.nips.cc/paper/5484-do-
deep-nets-really-need-to-be-deep.pdf](http://papers.nips.cc/paper/5484-do-deep-nets-really-need-to-be-deep.pdf) [abgerufen
am 23.03.2018]**

**Baram, Nir; Ansel, Oron; Mannor, Shie:
Model-based Adversarial Imitation Learning.
In: Computing Research Repository (CoRR),
07.12.2016, S. 1-9. [https://arxiv.org/abs/
1612.02179v1](https://arxiv.org/abs/1612.02179v1) [abgerufen am 22.03.2018]**

**Ho, Jonathan; Ermon, Stefano: Generative
Adversarial Imitation Learning. In: Computing
Research Repository (CoRR), 10.06.2016, S. 1-14.
<https://arxiv.org/abs/1606.03476v1> [abgerufen am
22.03.2018]**

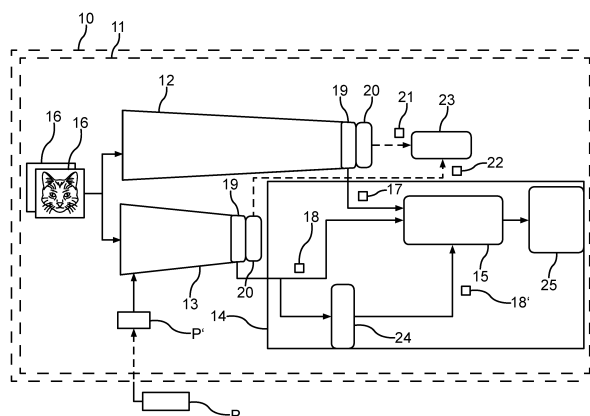
**Papernot, Nicolas [u.a.]: Semi-supervised
Knowledge Transfer for Deep Learning from
Private Training Data. In: Computing Research
Repository (CoRR), 03.03.2017, S. 1-16. [https://
arxiv.org/abs/1610.05755v4](https://arxiv.org/abs/1610.05755v4) [abgerufen am
22.03.2018]**

Rechercheantrag gemäß § 43 PatG ist gestellt.

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen.

(54) Bezeichnung: **VERFAHREN ZUM ERZEUGEN EINER KOMPRIMIERTEN VERSION EINES KÜNSTLICHEN NEURONALEN NETZWERKS SOWIE VORRICHTUNG UND SPEICHERMEDIUM**

(57) Zusammenfassung: Die Erfindung betrifft Verfahren zum Erzeugen einer komprimierten Version eines künstlichen neuronalen Netzwerks (12). Die Erfindung sieht vor, das Netzwerk (12) als ein Lehrernetzwerk (12) zusammen mit einem künstlichen neuronalen Diskriminatornetzwerk (15) und einem künstlichen neuronalen Schülernetzwerk (13), die beide als GANs, Generative Neural Networks, angeordnet sind, betrieben wird, wobei eine Konfiguration des Lehrernetzwerks (12) unverändert beibehalten wird und das Schülernetzwerk (13) mittels einer vorbestimmten GAN-Methode (14) darauf trainiert wird, zu mehreren vorbestimmten Eingabedatensätzen (16) eine jeweilige Ausgabe wie das Lehrernetzwerk (12) zu erzeugen, wobei das Diskriminatornetzwerk (15) als Diskriminator der GAN-Methode (14) betrieben wird, und das trainierte Schülernetzwerk (13) als die komprimierte Version des Lehrernetzwerks (12) bereitgestellt wird.



Beschreibung

[0001] Die Erfindung betrifft ein Verfahren zum Erzeugen einer komprimierten Version eines künstlichen neuronalen Netzwerks. Die komprimierte Version benötigt beispielsweise weniger Speicherplatz und/oder weniger Berechnungsschritte als das ursprüngliche Netzwerk, um zu einem Eingabedatensatz eine Ausgabe zu erzeugen. Die komprimierte Version des Netzwerks kann beispielsweise an eine vorbestimmte Hardware angepasst sein, für welche das ursprüngliche neuronale Netzwerk zu groß oder komplex ist. Zu der Erfindung gehören auch eine Vorrichtung und ein Speichermedium, mittels welchen das erfindungsgemäße Verfahren umgesetzt werden kann.

[0002] Ein künstliches neuronales Netzwerk kann beispielsweise für eine Bildverarbeitung genutzt werden. Besonders interessiert ist man daran, eine solche Bildverarbeitung in Echtzeit zu erhalten, was bedeutet, dass zwischen der Bereitstellung eines Bildes und einer zugehörigen Ausgabe des neuronalen Netzwerks höchstens eine vorbestimmte Höchstzeitdauer vergeht (sogenannte Echtzeitbedingung), wobei die Höchstzeitdauer kleiner als eine Sekunde sein kann oder noch vor oder mit dem Eintreffen des nächsten Bildes endet. Ein neuronales Netzwerk kann beispielsweise in einer Version als sogenanntes tiefes neuronales Netzwerk eine jeweilige Anzahl von Schichten und künstlichen Neuronen pro Schicht umfassen, die zu groß ist, um durch eine vorgegebene Prozessorhardware unter Einhaltung der Echtzeitbedingung ausgeführt werden zu können. Ein Beispiel hierzu ist eine Prozessorhardware einer Infrastrukturkomponente, die in einem Straßennetz installiert werden kann und mittels einer Kamera eine Verkehrsbeobachtung mit zugehöriger Bildauswertung bereitstellen kann. Wird hierbei ein neuronales Netzwerk verwendet, so muss dieses unter Umständen für die Prozessorhardware der Infrastrukturkomponente komprimiert werden, damit es weniger Schichten und/oder weniger Neuronen pro Schicht oder allgemein weniger Netzwerkparameter umfasst, damit diese beispielsweise in den Datenspeicher der Prozessorhardware passen und/oder unter Einhaltung einer Echtzeitbedingung bearbeitet werden können.

[0003] Ein neuronales Netzwerk wird zum Anpassen an eine vorbestimmte Aufgabe durch ein sogenanntes Training konfiguriert. Bei einem Training gibt man einem künstlichen neuronalen Netzwerk in einzelnen Trainingsschritten jeweils einen Eingabedatensatz, beispielsweise ein Bild oder ein Video, vor und vergleicht die Ausgabe des Netzwerks mit einer Soll-Ausgabe, d.h. dem zu dem Bild oder Video oder allgemein dem Eingabedatensatz gewünschten Klassifikationsergebnis, welches durch das Netzwerk erzeugt werden soll (Trainingsziel). Dann berechnet man einen Unterschied zwischen der tatsächlichen

Ausgabe des Netzwerks und dem gewünschten Klassifikationsergebnis und ändert daraufhin mittels einer sogenannten Loss-Funktion zumindest einen Konfigurationsparameter des Netzwerks in Abhängigkeit von dem berechneten Unterschied.

[0004] Eine hierbei bewährte Trainings-Methode ist die GAN-Methode, die zwei künstliche neuronale Netzwerke als sogenannte GANs (Generative Adversarial Networks) betreibt, wobei eines der beiden Netzwerke als Schülernetzwerk und das andere Netzwerk als Diskriminatornetzwerk bezeichnet wird. Die beiden Netzwerke werden mit entgegengesetzten Trainingszielen trainiert. Das Diskriminatornetzwerk wird nämlich dahingehend trainiert, eine Ausgabe des Schülernetzwerks als solche zu erkennen und nicht mit einer vorgegebenen Menge von Vergleichsausgaben zu verwechseln. Das Schülernetzwerk wird entgegengesetzt dazu dahingehend trainiert, Ausgaben zu erzeugen, die das Diskriminatornetzwerk nicht von den Vergleichsausgaben unterscheiden kann. Eine „Ausgabe“ des Schülernetzwerks ist jeweils ein Datensatz, den das Schülernetzwerk ausgibt.

[0005] Der Erfindung liegt die Aufgabe zugrunde, ein künstliches neuronales Netzwerk zu komprimieren, d.h. eine komprimierte Version dieses Netzwerks bereitzustellen.

[0006] Die Aufgabe wird durch die Gegenstände der unabhängigen Patentansprüche gelöst. Vorteilhafte Ausführungsformen der Erfindung ergeben sich durch die abhängigen Patentansprüche, die folgende Beschreibung sowie die Figuren.

[0007] Durch die Erfindung ist ein Verfahren zum Erzeugen einer komprimierten Version eines künstlichen neuronalen Netzwerks bereitgestellt. Dieses zu komprimierende Netzwerk wird hier als Lehrernetzwerk bezeichnet und zusammen mit einem künstlichen neuronalen Diskriminatornetzwerk und einem künstlichen neuronalen Schülernetzwerk betrieben, wobei diese beide in der beschriebenen Weise als GANs (Generative Adversarial Networks) angeordnet sind. Es werden also drei künstliche neuronale Netzwerke verwendet, das vorgegebene Netzwerk, das hier im Weiteren als Lehrernetzwerk bezeichnet ist, sowie ein Diskriminatornetzwerk und ein Schülernetzwerk, die beide zusammen GANs darstellen. Bei dem Verfahren wird eine Konfiguration des Lehrernetzwerks unverändert beibehalten, d.h. die Konfigurationsparameter bleiben unverändert. Das Lehrernetzwerk ist also fix oder konstant. Das Schülernetzwerk wird mittels einer vorbestimmten GAN-Methode darauf trainiert, zu mehreren vorbestimmten Eingabedatensätzen eine jeweilige Ausgabe wie das Lehrernetzwerk zu erzeugen. Ein solcher Eingabedatensatz kann beispielsweise jeweils ein Bild oder ein Video repräsentieren. Ein solcher Eingabedaten-

satz kann dem Schülernetzwerk und auf dem Lehrernetzwerk jeweils als Eingabe oder Input vorgegeben werden. In an sich bekannter Weise kann dann mittels einer sogenannten Loss-Funktion der GAN-Methode ein Training des Schülernetzwerks durchgeführt werden, um in der beschriebenen Weise mittels des Schülernetzwerks eine jeweilige Ausgabe zu den Eingabedatensätzen zu erzeugen, die der jeweiligen Ausgabe des Lehrernetzwerks entsprechen, d.h. von dem Diskriminatornetzwerk nicht mehr von einer Ausgabe des Lehrernetzwerks unterschieden werden können. Das Training erfolgt hier aber mittels einer GAN-Methode, d.h. das Diskriminatornetzwerk wird hierbei als Diskriminator der GAN-Methode in der beschriebenen Weise betrieben. Mit anderen Worten soll das Diskriminatornetzwerk durch sein Training unterscheiden können, ob die Ausgabe zu einem Eingabedatensatz von dem Schülernetzwerk oder dem Lehrernetzwerk kommt. Damit ergeben sich die beschriebenen entgegengesetzten Trainingsziele, indem das Schülernetzwerk erfolgreich versuchen muss, das Diskriminatornetzwerk zu täuschen (eine Ausgabe des Schülernetzwerks wird vom Diskriminatornetzwerk für eine Ausgabe des Lehrernetzwerks gehalten), während Diskriminatornetzwerk die Ausgaben des Schülernetzwerks als solche erkennen muss. Das Training kann hierbei iterativ erfolgen, d.h. in aufeinanderfolgenden Trainingsschritten, wobei jedem Trainingsschritt beispielsweise ein Eingabedatensatz Datensatz zu Grunde gelegt werden kann.

[0008] Die hierbei für die GAN-Methode verwendete Loss-Funktion kann beispielsweise eine Anpassung von Konfigurationsparametern des Schülernetzwerks vorsehen, wenn das Diskriminatornetzwerk erfolgreich eine Ausgabe des Schülernetzwerks als solche erkennt oder entlarvt hat, während eine Anpassung von Konfigurationsparametern des Diskriminatornetzwerks vorgesehen sein kann, wenn das Schülernetzwerk erfolgreich mit seiner Ausgabe das Diskriminatornetzwerk täuscht.

[0009] Durch das Trainieren des Schülernetzwerks auf Grundlage des Lehrernetzwerks und der GAN-Methode ergibt sich der Vorteil, dass eine in dem Lehrernetzwerk gespeicherte Information kompakt auf das Schülernetzwerk übertragen wird und hierdurch das Schülernetzwerk effizienter und/oder kompakter dieselbe Information anlegen oder speichern kann, als es das Lehrernetzwerk konnte, welches auf der Grundlage einer Vielzahl von Trainingsdatensätzen zunächst einmal die Information ermitteln musste.

[0010] Zu der Erfindung gehören auch Ausführungsformen, durch die sich zusätzliche Vorteile ergeben.

[0011] Eine Ausführungsform sieht vor, dass durch die GAN-Methode zu den mehreren Eingabedatensätzen, also beispielsweise zu mehreren Bildern oder

Videos, jeweils ein Lehrerausgabedatensatz durch das Lehrernetzwerk und ein Schülerausgabedatensatz durch das Schülernetzwerk als jeweilige Ausgabe erzeugt wird. Die Lehrerausgabedatensätze werden als solche deklariert oder gelabelt. Mit anderen Worten wird gegenüber dem Diskriminatornetzwerk kenntlich gemacht, dass es sich um Lehrerausgabedatensätze handelt, d.h. um eine jeweilige Ausgabe des Lehrernetzwerks. Die Schülerausgabedatensätze werden jeweils verdoppelt, d.h. kopiert. Von den Schülerausgabedatensätzen wird dabei jeweils eine Kopie als Lehrerausgabedatensatz umdeklariert oder vorgetäuscht und die andere, weitere Kopie korrekt als Schülerausgabedatensatz deklariert. Dem Diskriminatornetzwerk kann nun auf Grundlage der korrekt deklarierten Lehrerausgabedatensätze und beispielsweise der korrekt deklarierten Schülerausgabedatensätzen antrainiert oder beigebracht werden, wie ein Lehrerausgabedatensatz sowie ein Schülerausgabedatensatz aussieht. Des Weiteren muss dann das Diskriminatornetzwerk noch entscheiden, ob der ihm als Lehrerausgabedatensatz präsentierte, umdeklarierte Schülerausgabedatensatz von dem Lehrernetzwerk oder dem Schülernetzwerk stammt. Dem Diskriminatornetzwerk wird nicht verraten oder angezeigt oder signalisiert, dass es sich um einen Ausgabedatensatz handelt, der umdeklariert ist. Das Diskriminatornetzwerk wird also auf der Grundlage der Lehrerausgabedatensätze und der korrekt deklarierten und/oder der umdeklarierten Schülerausgabedatensätzen insgesamt darauf trainiert, die umdeklarierten Schülerausgabedatensätze trotz ihrer Umdeklaration jeweils korrekt als Schülerausgabedatensatz (und eben nicht als Lehrerausgabedatensatz) zu klassifizieren. Mit anderen Worten wird das Diskriminatornetzwerk darauf trainiert, Ausgaben des Lehrernetzwerks und Ausgaben des Schülernetzwerks jeweils unabhängig von deren Deklaration korrekt zu klassifizieren. Hierzu kann eine Loss-Funktion bereitgestellt werden, die eine umso größere Anpassung oder Veränderung von Konfigurationsparametern des Diskriminatornetzwerks vorsieht, je stärker oder mehr eine Ausgabe des Diskriminatornetzwerks einen umdeklarierten Schülerausgabedatensatz als Lehrerausgabedatensatz klassifiziert. Das Schülernetzwerk wird im Gegensatz dazu auf der Grundlage des Klassifizierungsergebnisses, welches das Diskriminatornetzwerk zu den umdeklarierten Schülerausgabedatensätzen erzeugt, also zu den gefälschten Ausgabedatensätzen, darauf trainiert, Schülerausgabedatensätze zu erzeugen, die das Diskriminatornetzwerk (zumindest nach der Umdeklaration) als Lehrerausgabedatensätze (also als Ausgaben des Lehrernetzwerks) klassifiziert, also falsch einordnet oder falsch klassifiziert. Für das Training des Schülernetzwerks kann eine Loss-Funktion verwendet werden, die eine umso größere Veränderung von Konfigurationsparametern des Schülernetzwerks vorsieht, je stärker oder mehr das Diskriminatornetzwerk die Schülerausgabedatensätze korrekt als sol-

che und nicht als Lehrerausgabedatensätze klassifiziert. Dem Begriff „stärker oder mehr“ ist hierbei eine Klassifikation zugrundegelegt, die keinen binären Wert (ja oder nein), sondern ein Intervall, beispielsweise einen Wahrscheinlichkeitswert oder einen logarithmischen Teilwahrscheinlichkeitswert, ausgibt, so dass das Klassifikationsergebnis eine Gradangabe zu einem der beiden möglichen Klassifikationsergebnisse darstellt, also z.B. das Klassifikationsergebnis „30% Wahrscheinlichkeit, dass es ein Schülerausgabedatensatz ist“. Das Training kann wieder iterativ erfolgen, d.h. pro Trainingsschritt kann ein Eingabedatensatz zu Grunde gelegt werden. Es ergibt sich der Vorteil, dass das Diskriminatornetzwerk sowohl mit den richtigen Lehrerausgabedatensätzen trainiert wird und hierdurch Lehrerausgabedatensätze als solche zu erkennen trainiert wird, als auch gleichzeitig oder im selben Trainingsschritt das Diskriminatornetzwerk und das Schülernetzwerk trainiert werden. Hierdurch kann ein vollständig und trainiertes Diskriminatornetzwerk und ein vollständig untrainiertes Schülernetzwerk als Grundlage für das Verfahren verwendet werden.

[0012] Eine Ausführungsform sieht vor, dass zum Trainieren des Schülernetzwerks nur die Klassifizierungsergebnisse zu den umdeklarierten Schülerausgabedatensätzen verwendet werden. Mit „nur“ ist gemeint, dass aus der Menge der Klassifizierungsergebnisse, die das Diskriminatornetzwerk zu den Lehrerausgabedatensätzen und den korrekt deklarierten und den umdeklarierten Schülerausgabedatensätzen erzeugt, nur die Klassifizierungsergebnisse zu den umdeklarierten Schülerausgabedatensätzen verwendet werden. Es hat sich gezeigt, dass dies das Training des Schülernetzwerks in vorteilhafter Weise besonders effizient macht.

[0013] Bisher wurde bei den Lehrerausgabedatensätzen und von den Schülerausgabedatensätzen jeweils von der jeweiligen „Ausgabe“ des Schülernetzwerks und des Lehrernetzwerks gesprochen. Dies muss aber nicht die finale Ausgabe, also beispielsweise eine sogenannte Logit-Ausgabe sein. Die Logit-Ausgabe ist die Ausgabe, d.h. derjenige Ausgabedatensatz, wie ihn die letzte Schicht oder Ausgabeschicht des jeweiligen Netzwerks erzeugt. Eine Ausführungsform sieht stattdessen vor, für die GAN-Methode als jeweilige Ausgabe des Schülernetzwerks eine Repräsentation einer vorbestimmten Zwischenschicht des Schülernetzwerks, d.h. einen entsprechenden Datensatz, der den Zustand der Zwischenschicht repräsentiert, und als jeweilige Ausgabe des Lehrernetzwerks eine Repräsentation einer vorbestimmten Zwischenschicht des Lehrernetzwerks, d.h. ein Datensatz, der den Zustand dieser Zwischenschicht repräsentiert, verwendet wird. Es hat sich in Versuchen herausgestellt, dass dies zu einem effizienteren Training des Schülernetzwerks führt.

[0014] Eine Ausführungsform sieht vor, dass das Schülernetzwerk im Vergleich zum Lehrernetzwerk mit einer gleichen oder einer kleineren Schichtanzahl und/oder mit weniger künstlichen Neuronen und/oder weniger künstlichen Neuronenverbindungen in zumindest einer Schicht bereitgestellt wird. Mit anderen Worten sind diese Netzwerkparameter (jeweilige Anzahl) kleiner gewählt. Hierdurch benötigt das Schülernetzwerk weniger Speicherplatz als das Lehrernetzwerk.

[0015] Zu beachten ist, dass hier zwischen Netzwerkparametern des Netzwerks und den Konfigurationsparametern unterschieden wird. Die Netzwerkparameter legen die Struktur des jeweiligen neuronalen Netzwerks fest (Schichtanzahl, Neuronenanzahl, Verbindungsanzahl). Die Konfigurationsparameter werden dagegen durch das Training eingestellt, um die einzutrainierende Information in dem Netzwerk zu speichern. Es kann sich z.B. jeweils um einen Gewichtungswert und/oder einen multiplikativen Faktor und/oder einen Wahrscheinlichkeitswert in einem der Neuronen oder einer der Neuronenverbindungen handeln. Ein Konfigurationsparameter stellt also einen Eigenschaftswert eines jeweiligen künstlichen Neurons und/oder einer künstlichen Neuronenverbindung dar. Er kann also ein Neuron oder eine Neuronenverbindung repräsentieren. Die Netzwerkparameter selbst (Anzahl an Schichten und/oder an Neuronen und/oder an Neuronenverbindungen) können dagegen beim Training unverändert bleiben.

[0016] Das Verändern der eigentlichen Netzwerkparameteranzahl im Schülernetzwerk (Änderung der Struktur) kann aber auch nachträglich erreicht werden. Eine Ausführungsform hierzu sieht vor, dass das Training des Schülernetzwerks ein Löschen von Neuronenverbindungen umfasst. Das Löschen kann in Abhängigkeit von zumindest einem Eigenschaftswert des Schülernetzwerks geschehen. Beispielsweise kann ermittelt werden, wie häufig eine Neuronenverbindung verwendet wird und/oder welchen Beitrag sie zu der Ausgabe liefert. Ist dies jeweils unter einem vorbestimmten Schwellenwert, so kann die Neuronenverbindungen gelöscht werden.

[0017] Eine Ausführungsform sieht vor, dass das Training des Schülernetzwerks zusätzlich umfasst, dass ein Unterschied zwischen einer zu den Eingabedatensätzen jeweils erzeugten Logit-Ausgabe des Lehrernetzwerks (also einer Ausgabe der letzten Schicht, d.h. der Ausgabeschicht des Lehrernetzwerks) und einer korrespondierende Logit-Ausgabe des Schülernetzwerks berechnet oder ermittelt wird und auf der Basis des Unterschieds eine zusätzliche Loss-Funktion für das Training angewendet wird. Diese Loss-Funktion kann beispielsweise eine Minimierung eines vorbestimmten Abstandsmaßes, mit welchem der Unterschied bestimmt oder berechnet wird, vorsehen. Diese weitere Loss-Funktion und die

Loss-Funktion der GAN-Methode können beispielsweise addiert oder gewichtet addiert werden. Die Gewichtung (ein multiplikativer Faktor) kann hierbei das Verhältnis oder den Einfluss der jeweiligen Loss-Funktion einstellen.

[0018] Eine Ausführungsform sieht vor, dass das Schülernetzwerk ohne gelabelte Trainingsdaten trainiert wird. Mit anderen Worten werden ausschließlich die Klassifizierungsergebnisse des Diskriminatornetzwerks und gemäß der beschriebenen Ausführungsform noch der Unterschied zu der Logit-Ausgabe des Lehrernetzwerks verwendet, um das Schülernetzwerk zu trainieren. Somit ist die Beschaffung von Trainingsdaten in vorteilhafter Weise nicht notwendig.

[0019] Eine Ausführungsform sieht vor, dass das Training des Schülernetzwerks umfasst, dass in einem oder mehreren oder allen Trainingsschritten des Trainings einige Konfigurationsparameter des Schülernetzwerks konstant gehalten werden. Mit „konstant gehalten“ ist gemeint, dass diese Konfigurationsparameter ansonsten in den Trainingsschritt aufgrund der GAN-Methode und/oder der beschriebenen zusätzlichen Loss-Funktion geändert würden. Es hat sich herausgestellt, dass diese Ausführungsform ein sogenanntes Overfitting des Schülernetzwerks verhindert.

[0020] Eine Ausführungsform sieht vor, dass jeweilige initiale Konfigurationsparameter des Schülernetzwerks und des Diskriminatornetzwerks jeweils mit Zufallszahlen oder mit einem vorbestimmten, von dem Lehrernetzwerk unabhängigen Zahlenmuster initialisiert werden. Hierdurch können sich die GANs besonders flexibel an das Lehrernetzwerk anpassen. Es kann verhindert werden, dass das Schülernetzwerk oder das Diskriminatornetzwerk eine anfängliche Tendenz oder einen anfänglichen Trend aufweist.

[0021] Eine Ausführungsform sieht vor, dass als das Lehrernetzwerk ein tiefes neuronales Netzwerk mit einer oder zwei oder mehr als zwei Zwischenschichten bereitgestellt wird. Bei einem solchen tiefen neuronalen Netzwerk ergibt das Komprimieren einen besonders hohen Kompressionsgewinn.

[0022] Eine Ausführungsform sieht vor, dass eine Schichtanzahl und/oder eine Neuronenanzahl und/oder eine Anzahl an Neuronenverbindungen in dem Schülernetzwerk, d.h. also zumindest ein Netzwerkparameter des Schülernetzwerks (Anzahl an Schichten und/oder künstlichen Neuronen und/oder an Neuronenverbindungen) in Abhängigkeit von einer vorbestimmten Prozessorhardware angepasst wird. Mit anderen Worten wird das zu komprimierende künstliche neuronale Netzwerk (Lehrernetzwerk) dahingehend komprimiert, dass die Prozessorhardware be-

rücksichtigt wird, damit also die komprimierte Version beispielsweise in einen Datenspeicher der Prozessorhardware passt und/oder von der Prozessorhardware berechnet oder betrieben werden kann.

[0023] Insbesondere ist dabei gemäß einer Ausführungsform vorgesehen, dass das Anpassen dahingehend Erfolg, dass das Schülernetzwerk in Unterschied zum Lehrernetzwerk eine vorbestimmte Echtzeitbedingung erfüllt, wenn es durch die Prozessorhardware betrieben wird. Mit anderen Worten wird die Schichtanzahl und/oder die Netzwerkparameteranzahl Schülernetzwerks dahingehend eingestellt, dass die Prozessorhardware das Schülernetzwerk unter Erfüllung der Echtzeitbedingung berechnen oder betreiben kann, während dies mit dem Lehrernetzwerk unmöglich ist. Die Echtzeitbedingung kann beispielsweise besagen, dass zu einem Eingabedatensatz Logit-Ausgabe oder die Ausgabe der Ausgabeschicht des Schülernetzwerks innerhalb einer vorbestimmten Zeitdauer erzeugt oder berechnet werden kann. Die Zeitdauer kann in einem Bereich kleiner als 10 Sekunden, insbesondere kleiner als 5 Sekunden, insbesondere kleiner als 1 Sekunde, liegen. Die Zeitdauer kann kleiner oder gleich einer Rate sein, mit welcher in der vorgesehenen Anwendung Eingabedatensätze (z.B. Bilder oder Videos) erzeugt werden. Das Anpassen an die Echtzeitbedingung kann erfolgen, indem ermittelt wird, wie viele Berechnungen die Proessoreinrichtung pro Zeiteinheit, z.B. pro Sekunde, durchführen kann und wie viele Berechnungen ein künstliches Neuron und/oder eine künstliche Neuronenverbindung benötigt. Daraus kann eine zulässige Höchstanzahl an Netzwerkparametern des Schülernetzwerks ermittelt werden.

[0024] Eine Ausführungsform sieht vor, dass das Netzwerk, welches als Lehrernetzwerk verwendet wird, also das zu komprimierende Netzwerk, vor der Verwendung als Lehrernetzwerk für eine der folgenden Funktionen Mustererkennung, Bildverarbeitung, Objektklassifikation (in Sensordaten abgebildeter Objekte), Verkehrssituationserkennung (zum Beispiel Erkennung von Stau und/oder stockendem Verkehr und/oder freier Fahrstrecke) oder Verkehrsobjekterkennung (zum Beispiel Erkennung von Fußgängern und/oder Kraftfahrzeugen und/oder Radfahrern) trainiert wird und diese Funktion nach dem Training aber dann mittels des Schülernetzwerks durchgeführt oder ausgeführt wird. Somit kann die Funktion zunächst dem Lehrernetzwerk antrainiert werden und dann aber mit geringerem Prozessoraufwand durch das Schülernetzwerk durchgeführt werden.

[0025] Gemäß einer Ausführungsform wird hierbei das Schülernetzwerk in einer Verkehrsbeobachtungseinrichtung für eine Verkehrssituationserkennung oder Verkehrsobjekterkennung betrieben, wobei diese Verkehrsbeobachtungseinrichtung in einer Infrastrukturkomponente einer Straßeninfrastruktur

tur installiert ist. Eine Straßeninfrastruktur ist ein Straßennetz. Eine Infrastrukturkomponente kann beispielsweise durch eine Brücke und/oder einen Mast (Laternenmast oder Schildermast) und/oder durch eine Leitplanke und/oder eine am Straßenrand installierte elektronische Einrichtung realisiert sein.

[0026] Um das erfindungsgemäße Verfahren durchzuführen, ist durch die Erfindung auch eine Vorrichtung mit einer Prozessoreinrichtung bereitgestellt, wobei die Prozessoreinrichtung dazu eingerichtet ist, eine Ausführungsform des erfindungsgemäßen Verfahrens durchzuführen. Die Prozessoreinrichtung kann hierzu zumindest einen Mikroprozessor aufweisen. Die Prozessoreinrichtung kann einen Programmcode aufweisen, der Programminstruktionen oder Programmbefehle aufweist, die dazu eingerichtet sind, die Ausführungsform erfindungsgemäßen Verfahrens durchzuführen. Der Programmcode kann in einem Datenspeicher der Vorrichtung gespeichert sein.

[0027] Um einen solchen Datenspeicher zu realisieren, sieht die Erfindung ein Speichermedium mit dem darin gespeicherten Programmcode vor, wobei dieser Programmcode in der beschriebenen Weise dazu eingerichtet ist, bei Ausführen durch eine Prozessoreinrichtung eine Ausführungsform des erfindungsgemäßen Verfahrens durchzuführen. Das Speichermedium kann beispielsweise als eine Festplatte oder als eine DVD oder als ein Datenspeicher eines Computers oder eines Verbunds aus mehreren Computern realisiert sein.

[0028] Die Erfindung umfasst auch die Kombinationen der beschriebenen Ausführungsformen.

[0029] Im Folgenden ist ein Ausführungsbeispiel beschrieben. Hierzu zeigt:

Fig. 1 eine schematische Darstellung einer Ausführungsform der erfindungsgemäßen Vorrichtung; und

Fig. 2 ein Flusschaudiagramm zu einem Verfahren, das durch die Vorrichtung gemäß **Fig. 1** ausgeführt werden kann;

Fig. 3 eine schematische Darstellung eines Trainings eines Lehrernetzwerks; und

Fig. 4 eine schematische Darstellung zu einer Verwendung eines von der Vorrichtung erzeugten Schülernetzwerks in einer Verkehrsbeobachtungseinrichtung.

[0030] In den Figuren sind funktionsgleiche Elemente jeweils mit denselben Bezugszeichen versehen.

[0031] **Fig. 1** zeigt eine Vorrichtung **10**, die beispielsweise einen Computer oder einen Verbund aus Computern umfassen kann. Die Vorrichtung **10** kann eine

Prozessoreinrichtung **11** aufweisen, mittels welcher das im folgenden beschriebene Verfahren zum Komprimieren eines künstlichen neuronalen Netzwerks durchgeführt werden, das bei dem Verfahren als Lehrernetzwerk **12** betrieben wird. Bei dem Verfahren bleibt das Lehrernetzwerk **12** untrainiert oder mit anderen Worten werden dessen Konfigurationsparameter unverändert beibehalten. Die komprimierte Version des Lehrernetzwerks **12** wird mittels eines Schülernetzwerks **13** bereitgestellt, das ebenfalls ein neuronales Netzwerk ist, wobei aber beispielsweise eine Anzahl an Zwischenschichten und/oder allgemein eine Anzahl an Netzwerkparametern, beispielsweise eine Anzahl an künstlichen Neuronen und/oder Netzwerkverbindungen, kleiner als bei dem Lehrernetzwerk **12** sein kann.

[0032] Parameter des Schülernetzwerks **13**, beispielsweise eine Anzahl an Zwischenschichten und/oder eine Anzahl an künstliche Neuronen und/oder Neuronenverbindungen in zumindest einer oder mehreren oder allen der Schichten des Schülernetzwerks **13**, kann durch eine Prozessorkonfiguration **P'** festgelegt sein, die eine Prozessorhardware **P** beschreiben, durch welche das Schülernetzwerk **13** betrieben werden soll.

[0033] Das Übertragen der in dem Lehrernetzwerk **12** gespeicherten Informationen auf das Schülernetzwerk **13** erfolgt durch Trainieren des Schülernetzwerks **13**. Hierzu wird eine GAN-Methode **14** verwendet, für die ein weiteres künstliches neuronales Netzwerk, nämlich ein Diskriminatornetzwerk **15**, bereitgestellt wird.

[0034] Das Verfahren ist im Folgenden auf Grundlage von **Fig. 1** und **Fig. 2** beschrieben. Das Verfahren kann zyklisch durchgeführt werden, wodurch sich ein interaktives Training des Schülernetzwerks **13** ergibt. In einem Schritt **S10** kann hierbei in der beschriebenen Weise die Konfiguration des Lehrernetzwerks **12** beibehalten oder fixiert bleiben. In einem Schritt **S11** können in jedem Trainingsschritt zu jeweils einem Eingabedatensatz **16** aus mehreren Eingabedatensätzen **16** durch das Lehrernetzwerk **12** und dem Schülernetzwerk **13** jeweils eine Ausgabe erzeugt werden, wobei diese Ausgabe bei dem Lehrernetzwerk **12** ein Lehrerausgabedatensatz **17** und bei dem Schülernetzwerk **13** ein Schülerausgabedatensatz **18** ist. Die Ausgabe kann hierbei den aktuellen Zustand einer jeweiligen Zwischenschicht **19** repräsentieren oder darstellen. Zusätzlich kann in einem Schritt **S12** von dem Lehrernetzwerk **12** und den Schülernetzwerk **13** auch die Ausgabe der Ausgabeschicht **20**, d.h. die Logit-Ausgabe **21** des Lehrernetzwerks **12** und die Logit-Ausgabe **22** des Schülernetzwerks **13**, erfasst werden. Die Logit-Ausgaben **21**, **22** können für eine weitere Loss-Funktion **23** (L2) verwendet werden, um die zukünftigen Logit-Ausgaben **22** des Schülernetzwerks **13** an die korrespondieren-

den, zum selben Eingabedatensatz **16** erzeugten Logit-Ausgaben **21** des Lehrernetzwerks **12** anzugleichen.

[0035] Bei der GAN-Methode **14** werden zum einen das Diskriminatornetzwerk **15** und zum anderen das Schülernetzwerk **13** trainiert. Hierzu wird in jedem Trainingsschritt der Lehrerausgabedatensatz **17** und der Schülerausgabedatensatz **18** jeweils als solcher in einem Schritt **S13** deklariert oder markiert oder gelabelt. Zusätzlich wird eine Kopie des Schülerausgabedatensatzes **18** in einem Schritt **S14** durch eine Umdeklaration **24** zu einem umdeklarierten Schülerausgabedatensatz **18'** umdeklariert oder umgelabelt, sodass dem Diskriminatornetzwerk **15** dieser umdeklarierte Schülerausgabedatensatz **18'** mit einem Label oder einer Deklaration als Lehrerausgabedatensatz präsentiert wird. Das Diskriminatornetzwerk **15** wird nun in einem Schritt **S14** zum einen darauf trainiert, zwischen einem Lehrerausgabedatensatz **17** und einem Schülerausgabedatensatz **18** zu unterscheiden und zum anderen in einer Klassifizierung **25** anzugeben, ob es bei dem umdeklarierten Schülerausgabedatensatz **18'** um einen Schülerausgabedatensatz oder einen Lehrerausgabedatensatz handelt. Abhängigkeit von dieser Klassifizierung **25** kann beispielsweise mittels einer Loss-Funktion der GAN-Methode zusammen mit der Loss-Funktion **23** (**L2**) das Schülernetzwerk **13** in einem Schritt **S15** darauf trainiert werden, zukünftig Schülerausgabedatensätze **18** zu erzeugen, bei welchen das Diskriminatornetzwerk **15** die umdeklarierten Schülerausgabedatensätze **18'** fälschlicherweise als Lehrerausgabedatensatz klassifiziert.

[0036] Das Training erfolgt insbesondere mittels einer Back-Propagation Methode.

[0037] Bezeichnet man die Loss-Funktion für die GAN-Methode als **L1**, so können die beiden Loss-Funktionen z.B. wie folgt kombiniert werden:

$$\arg \min \max (L1 + I * L2)$$

wobei die Minimierung **min** bezüglich des Schülernetzwerks und die Maximierung **max** bezüglich des Diskriminatornetzwerks erfolgen soll und ***** die Multiplikation und **I** ein Tuning-Wert zum Mischen der beiden Loss-Funktionen **L1**, **L2** ist und vom Fachmann nach Bedarf eingestellt werden kann.

[0038] Es kann hierbei zunächst die Maximierung **max** bezüglich des Diskriminatornetzwerks erfolgen, d.h. es werden Werte von vorbestimmten Konfigurationsparametern des Diskriminatornetzwerks ermittelt, die zu einer Maximierung des obigen mathematischen Ausdrucks (**L1** + **I** * **L2**) hin führen. Die ermittelten Werte für die Konfigurationsparameter können dann in dem Diskriminatornetzwerk eingetragen

werden. Dann kann die Minimierung **min** bezüglich des Schülernetzwerks vorgenommen werden, d.h. es werden Werte für Konfigurationsparameter des Schülernetzwerks ermittelt, die zu einer Minimierung des obigen mathematischen Ausdrucks hin führen. Die ermittelten Werte für die Konfigurationsparameter können dann in dem Schülernetzwerk eingetragen werden.

[0039] Das Minimieren und das Maximieren kann durch Ermitteln der ersten und optional der zweiten mathematischen Ableitung des obigen mathematischen Ausdrucks erfolgen. Die Ableitung kann bezüglich der einzustellenden Konfigurationsparameter durchgeführt werden.

[0040] **L1** kann definiert sein als die Summe des statistischen Erwartungswerts **E** des Logarithmus der Ausgabe des Diskriminatornetzwerks **D** betreffend den jeweils in einem Trainingsschritt erzeugten Lehrerdatsatz **Ft** (das heißt $\log D(Ft)$) einerseits und des statistischen Erwartungswerts **E** des Logarithmus \log von 1 minus der Ausgabe **D(Fs)** des Diskriminatornetzwerks **D** zu dem umdeklarierten Schülerausgabedatensatz **Fs** des Trainingsschritts (das heißt $\log(1-D(Fs))$) andererseits:

$$L1 = E[\log D(Ft)] + E[\log(1 - D(Fs))].$$

[0041] **L2** kann definiert sein als der Erwartungswert eines Abstandsmaßes $\| \cdot \|^2$ (z.B. Betragsquadrat eines Vektors) zwischen der Logit-Ausgaben **ft** des Lehrernetzwerks **12** und der Logit-Ausgabe **fs** des Schülernetzwerks **13**:

$$L2 = E[\|ft - fs\|^2].$$

[0042] **Fig. 3** veranschaulicht, wie das Lehrernetzwerk **12** zuvor trainiert worden sein kann. Das Lehrernetzwerk **12** kann beispielsweise ein tiefes neuronales Netzwerk sein. In einem Training des Lehrernetzwerks kann auf Grundlage von Eingabedatensätzen **16** in jedem Trainingsschritt jeweils zu einem Eingabedatensatz **16** und einer zugehörigen gewünschten Soll-Klassifikation oder Label **26** eine an sich bekannte Trainingsmethode **27** für künstliche neuronale Netzwerke verwendet werden, um Konfigurationsdaten des Lehrernetzwerks **12** anzupassen.

[0043] **Fig. 4** veranschaulicht, wie das Schülernetzwerk **13** verwendet werden kann. Hierzu kann dank des Schülernetzwerks **13** die Prozessorhardware **P** benutzt werden, deren Konfiguration **P'** dem Schülernetzwerk **13** zugrundegelegt wurde.

[0044] Als Eingabedatensatz **16** kann beispielsweise aus einer Kamera **28** ein Bild **29** oder Video beispielsweise einer Verkehrsszene bereitgestellt wer-

den. Die Prozessorhardware **P** kann hierzu beispielsweise als Bestandteil einer Verkehrsbeobachtungseinrichtung in einer Infrastrukturkomponente, beispielsweise einer Brücke und/oder einer Straßenlaterne installiert sein. Das Schülernetzwerk **13** kann als Merkmalsextraktion **30** betrieben werden und hierzu in einem ersten Erkennungsdurchlauf in dem Bild **29** einzelne Verkehrsobjekte **31** markieren oder erkennen. Es ergeben sich Objektvorschläge **33**. Beispielsweise kann jedes Verkehrsobjekt **31** durch eine sogenannte Bounding-Box **32** markiert sein. Der Übersichtlichkeit halber sind in **Fig. 4** von den Verkehrsobjekten **31** und deren Bounding-Boxen **32** jeweils nur für einige Verkehrsobjekte mit Bezugszeichen versehen. Ein Verkehrsobjekt **31** kann beispielsweise ein Kraftfahrzeug und/oder ein anderer Verkehrsteilnehmer, beispielsweise ein Fußgänger oder Radfahrer, sein.

[0045] In einem weiteren Erkennungsschritt kann nun auf Grundlage der Bildausschnitte der einzelnen Bounding-Boxen **32** individuell für jedes Verkehrsobjekt **31** erneut eine Merkmalsextraktion mittels des Schülernetzwerks **13** durchgeführt werden und hierdurch eine individuelle, für jedes Verkehrsobjekt **31** vorgesehene Objektklassifikation **34** vorgenommen werden.

[0046] Auf Grundlage der Objektklassifikation **34** kann nun beispielsweise eine Anzahl der Verkehrsobjekte **31** in dem durch die Kamera **28** erfassten Bereich und/oder eine Art der Verkehrsobjekte **31** ermittelt werden. Aufgrund der Anzahl und/oder der Anordnung und/oder der Ausrichtung der Verkehrsobjekte **31** kann dann beispielsweise auf eine aktuelle Verkehrssituation (Stau und/oder stockender Verkehr und/oder fließender Verkehr) rückgeschlossen werden. Jedes Verkehrsobjekt **31** kann auch individuell erfasst werden, um beispielsweise zu erkennen, ob es sich um ein Kraftfahrzeug und/oder einen Fußgänger oder einen Radfahrer handelt.

[0047] Indem das Schülernetzwerk mittels einer GAN-Methode trainiert wurde, handelt es sich um eine Version des Lehrernetzwerks, die weniger Speicherplatz und/oder weniger Berechnungsschritte benötigt. Hierzu wird das beschriebene adversale Lernen der GAN-Methode verwendet. Es ergibt sich also eine adversale Kompression. Bei dieser diesem Verfahren wird Wissen oder Information von einem neuronalen Netzwerk mit großer Anzahl an Netzwerkparametern auf ein neuronales Netzwerk mit limitierter Anzahl an Netzwerkparametern übertragen. Das Lehrernetzwerk wird vorher der beschriebenen Weise mit gelabelten Trainingsdaten trainiert (siehe **Fig. 3**). Das Training des Lehrernetzwerks ist beendet, wenn es eine vorbestimmte Klassifizierungsgenauigkeit oder Erkennungsgenauigkeit aufweist. Danach kann das Schülernetzwerk darauf trainiert werden, dass Lehrernetzwerk nachzuahmen oder zu ge-

gebenen Eingabedatensätzen jeweils einen Logit-Ausgabe zu erzeugen, deren Unterschied zur Logit-Ausgabe des Lehrernetzwerks kleiner als ein vorbestimmter Schwellenwert ist. Das Verfahren bietet hierbei eine Möglichkeit, das Schülernetzwerk zu trainieren.

[0048] Es wird adversales Lernen für das Training des Schülernetzwerks verwendet. Dies bedeutet, dass ein weiteres künstliches neuronales Netzwerk, nämlich das Diskriminatornetzwerk, zusammen mit dem Schülernetzwerk darauf trainiert wird, zu unterscheiden oder zu erkennen, ob eine Ausgabe oder ein Trainingsampel von dem Lehrernetzwerk oder dem Schülernetzwerk stammt. Das Diskriminatornetzwerk empfängt hierzu als Eingabe die Ausgabe einer Zwischenschicht des Lehrernetzwerks oder des Schülernetzwerks (d.h. den Lehrerausgabedatensatz und den Schülerausgabedatensatz). Gleichzeitig wird mittels der zweiten oder zusätzlichen Loss-Funktion **L2** der Unterschied zwischen dem finalen Ausgaben oder Logit-Ausgaben des Lehrernetzwerks und des Schülernetzwerks minimiert. Hierdurch ergibt sich insgesamt die adversale Netzwerk-kompression. Das Training des Schülernetzwerks und des Diskriminatornetzwerks ist in **Fig. 1** zusammengefasst.

[0049] Durch das Verfahren ergeben sich drei wesentliche Vorteile. Es ist erstens generisch und kann somit auf jede neuronalen Netzwerk-Architektur angewendet werden und zwar unabhängig von dem Inhalt des Trainings dieses Netzwerks. Zum Beispiel kann es für jede beliebige Klassifikationsaufgabe und/oder die Objekterkennung verwendet werden, die Beispiele für wichtige Erkennungsaufgaben darstellen. Zweitens wird keine spezielle Hardware und/oder Programmbibliotheken und/oder Software benötigt. Drittens kann ein Overfitting durch ein sogenanntes Dropout-Training oder eine Dropout-Regularisierung erreicht werden, d.h. es werden in zumindest einem Trainingsschritt einige Konfigurationsparameter unverändert oder konstant beibehalten.

[0050] Im Folgenden sind zwei bevorzugte konkrete Anwendungen des Verfahrens beschrieben.

[0051] In einer ersten Ausführungsform wird als Lehrernetzwerk ein Convolutional Neural Network (CNN) für eine Bildklassifizierung von Vorgängen, beispielsweise einer Verkehrssituation, beschrieben. Hierzu wird auf **Fig. 4** Bezug genommen. Eine Kamera **28**, beispielsweise eine RGB-Kamera (RGB - rot grün blau), kann in einem Außenbereich wird draußen an einem Mast einer Laterne befestigt sein und tägliche Fahrzeug-basierte städtische Aktivitäten klassifizieren oder erkennen, um hierdurch eine Verkehrssituationserkennung, beispielsweise ein Grad oder Maß eines Verkehrsstaus, ein Fahrzeug mit vorbestimmter unerlaubter Fahrweise, einen Verkehrsunfall und/

oder das Queren der Straße durch Fußgänger zu erkennen. Um diese Verkehrssituationserkennung unter Einhaltung einer Echtzeitbedingung durchzuführen, kann ein hierfür trainiertes Lehrernetzwerk mittels des Verfahrens komprimiert werden. Das Lehrernetzwerk wird hierzu mit Trainingsdaten und Labels **26** zum Klassifizieren von Bildern **29** trainiert, um die entsprechende Kategorie der Verkehrssituation in jedem Bild **29** zu erkennen. Das Lehrernetzwerk kann hierzu ein tiefes neuronales Netzwerk sein, das aber zu umfangreich für die Prozessorhardware **P** ist, mit welcher die Verkehrssituationserkennung durchgeführt werden soll. Dafür kann es aufgrund seines Umfangs aber sehr genau trainiert werden. Danach können das Studentennetzwerk und das Diskriminatornetzwerk zusammen trainiert werden, um hierdurch das in dem Lehrernetzwerk gespeicherte Wissen oder die gespeicherte Information von dem Lehrernetzwerk auf das Schülernetzwerk zu übertragen. Das Lehrernetzwerk wird unverändert beibehalten und nur die beiden anderen Netzwerke (Schülernetzwerk und Diskriminatornetzwerk) werden trainiert, d.h. ihre Konfigurationsparameter sind veränderbar. Die Konfigurationsparameter des Schülernetzwerks und des Diskriminatornetzwerks werden mit Zufallszahlen initialisiert. Während des Trainings gibt es zwei Trainingsziele. Das erste ist die Minimierung des Unterschieds zwischen der Logit-Ausgabe des Lehrernetzwerks und der korrespondierenden Logit-Ausgabe des Schülernetzwerks (Loss-Funktion **L2**). Das zweite Trainingsziel ist die Minimierung der Kreuzentropie (Cross Entropie) für das Diskriminatornetzwerk, wenn das Diskriminatornetzwerk die Klassifizierung **25** durchführt, d.h. feststellt, ob eine Ausgabe von dem Lehrernetzwerk oder dem Schülernetzwerk stammt. Die Eingabe hierzu für das Diskriminatornetzwerk ist eine Ausgabe vom Lehrernetzwerk und die Ausgabe des Schülernetzwerks. Beim letzteren ist aber eine Kopie zusätzlich als umdeklierte Ausgabe (der umdeklierte Schülerausgabedatensatz **18'**) bereitgestellt. Somit ergeben sich insgesamt drei Ausgabedatensätze oder Samples in einem Trainingsschritt für das Diskriminatornetzwerk bei gegebenen Eingabedatensatz **16**. Durch die Umdeklaration **24** wird versucht, das Diskriminatornetzwerk zu täuschen. Hieran kann das Schülernetzwerk seine Konfigurationsparameter beispielsweise mit einem Gradienten aktualisieren, der dem Gradienten des Lehrernetzwerks für einen Lernschritt entsprechen würde. Schließlich lernt insgesamt hierdurch der Schülernetzwerk Logit-Ausgaben zu erzeugen, die bei gegebenen Eingabedatensätzen **16** denjenigen des Lehrernetzwerks entsprechen.

[0052] In der zweiten bevorzugten Ausführungsform wird ebenfalls ein Convolutional Neural Netzwerk als Lehrernetzwerk für die Verkehrsbeobachtung verwendet, und zwar für die Verkehrsobjekterkennung. Die Kamera **28** kann wieder in der beschriebenen Weise z.B. an einem Pfosten einer Laterne bereit-

gestellt sein. Ziel ist es, in den Bildern **29** der Kamera **28** einzelne Verkehrsobjekte **31** zu lokalisieren, bevorzugt unter Einhaltung einer Echtzeitbedingung. In der beschriebenen Weise können mittels der Merkmalsextraktion **30** die Objektvorschläge **33** erzeugt werden und dann für jeden einzelnen Objektvorschlag noch mal eine Merkmalsextraktion **30** zum Beschreiben beispielsweise der Raumlage und/oder der Art des Verkehrsobjekts **31** durchgeführt werden.

[0053] Beide Ausführungsformen können insbesondere dazu verwendet werden, autonomes Fahren für zumindest ein Kraftfahrzeug zu unterstützen. Insbesondere kann zumindest ein Kraftfahrzeug oder ein anderer Verkehrsteilnehmer und ein anderes Verkehrsobjekt **31** erkannt und dessen Aktivität oder Verhalten automatisiert erkannt oder klassifiziert werden und dann dem zumindest einen autonom fahrenden Kraftfahrzeug signalisiert werden. Hierbei ist es insbesondere vorteilhaft, dass das autonom fahrende Kraftfahrzeug keinen Sichtkontakt zu der jeweiligen Verkehrsszene haben muss, bis es dort eintrifft. Die Kamera **28** kann hierzu an einer Infrastrukturkomponente, wie beispielsweise einer Straßenlaterne, bereitgestellt sein und damit als entferntes Sensorsystem für das autonome Kraftfahrzeug betrieben werden.

Bezugszeichenliste

10	Vorrichtung
11	Prozessoreinrichtung
12	Lehrernetzwerk
13	Schülernetzwerk
14	GAN-Methode
15	Diskriminatornetzwerk
16	Eingabedatensatz
17	Lehrerausgabedatensatz
18	Schülerausgabedatensatz
18'	Umdeklierter Schülerausgabedatensatz
20	Ausgabeschicht
21	Logit-Ausgabe
22	Logit-Ausgabe
23	Loss-Funktion
24	Umdeklaration
25	Klassifikationen
26	Label
27	Trainingsziel
28	Kamera
39	Bild

30	Merkmalsextraktion
31	Verkehrsobjekt
32	Bounding-Box
33	Objektvorschlag
34	Objektklassifikation
P	Prozessorhardware
P'	Prozessorkonfiguration
S10-S15	Verfahrensschritt

Patentansprüche

1. Verfahren zum Erzeugen einer komprimierten Version eines künstlichen neuronalen Netzwerks (12), **dadurch gekennzeichnet**, dass das Netzwerk (12) als ein Lehrernetzwerk (12) zusammen mit einem künstlichen neuronalen Diskriminatornetzwerk (15) und einem künstlichen neuronalen Schülernetzwerk (13), die beide als GANs, Generative Neural Networks, angeordnet sind, betrieben wird, wobei

- eine Konfiguration des Lehrernetzwerks (12) unverändert beibehalten wird und
- das Schülernetzwerk (13) mittels einer vorbestimmten GAN-Methode (14) darauf trainiert wird, zu mehreren vorbestimmten Eingabedatensätzen (16) eine jeweilige Ausgabe wie das Lehrernetzwerk (12) zu erzeugen, wobei für das Training das Diskriminatornetzwerk (15) als Diskriminator der GAN-Methode (14) betrieben wird, und
- das trainierte Schülernetzwerk (13) als die komprimierte Version des Lehrernetzwerks (12) bereitgestellt wird.

2. Verfahren nach Anspruch 1, wobei durch die GAN-Methode (14) zu den mehreren Eingabedatensätzen (16)

- jeweils ein Lehrerausgabedatensatz (17) durch das Lehrernetzwerk (12) und ein Schülerausgabedatensatz (18) durch das Schülernetzwerk (13) als jeweilige Ausgabe erzeugt wird, und
- die Lehrerausgabedatensätze (17) als solche deklariert werden und
- von den Schülerausgabedatensätzen (18) jeweils eine Kopie als Lehrerausgabedatensatz umdeklariert und eine Kopie korrekt als Schülerausgabedatensatz (18) deklariert wird und
- das Diskriminatornetzwerk (15) auf der Grundlage der Lehrerausgabedatensätze (17) und der korrekt deklarierten und/oder der umdeklarierten Schülerausgabedatensätze (18, 18') darauf trainiert wird, die umdeklarierten Schülerausgabedatensätze (18') trotz ihrer Umdeklaration (24) korrekt als Schülerausgabedatensätze (18) zu klassifizieren, und
- das Schülernetzwerk (13) auf der Grundlage des Klassifizierungsergebnisses (25), welches das Diskriminatornetzwerk (15) zu den umdeklarierten Schülerausgabedatensätzen (18') erzeugt, darauf trainiert wird, Schülerausgabedatensätze (18) zu erzeugen,

die das Diskriminatornetzwerk (15) als Lehrerausgabedatensätze (17) klassifiziert.

3. Verfahren nach einem der vorhergehenden Ansprüche, wobei zum Trainieren des Schülernetzwerks (13) aus der Menge der Klassifizierungsergebnisse (25) zu den Lehrerdatsätzen (17) und den korrekt deklarierten und den umdeklarierten Schülerausgabedatensätzen (18, 18') nur die Klassifizierungsergebnisse (25) zu den umdeklarierten Schülerausgabedatensätzen (18') verwendet werden.

4. Verfahren nach einem der vorhergehenden Ansprüche, wobei als jeweilige Ausgabe des Schülernetzwerks (13) eine Repräsentation (19) einer vorbestimmten Zwischenschicht des Schülernetzwerks (13) und als jeweilige Ausgabe des Lehrernetzwerks (12) eine Repräsentation (19) einer vorbestimmten Zwischenschicht des Lehrernetzwerks (12) verwendet wird.

5. Verfahren nach einem der vorhergehenden Ansprüche, wobei das Schülernetzwerk (13) im Vergleich zum Lehrernetzwerk (12) mit einer gleichen oder einer kleineren Schichtanzahl und/oder mit weniger Netzwerkparametern, insbesondere weniger künstlichen Neuronen und/oder weniger künstlichen Neuronenverbindungen, in zumindest einer Schicht bereitgestellt wird.

6. Verfahren nach einem der vorhergehenden Ansprüche, wobei das Training des Schülernetzwerks (13) ein Löschen von Neuronenverbindungen umfasst.

7. Verfahren nach einem der vorhergehenden Ansprüche, wobei das Training des Schülernetzwerks (13) zusätzlich umfasst, dass ein Unterschied zwischen einer zu den Eingabedatensätzen (16) jeweils erzeugten Logit-Ausgabe (21) des Lehrernetzwerks (12) und einer Logit-Ausgabe (22) des Schülernetzwerks (13) ermittelt wird und auf Basis des Unterschieds eine zusätzliche Loss-Funktion (23) für das Training angewendet wird.

8. Verfahren nach einem der vorhergehenden Ansprüche, wobei das Schülernetzwerk (13) ohne gelabelte Trainingsdaten trainiert wird.

9. Verfahren nach einem der vorhergehenden Ansprüche, wobei das Training des Schülernetzwerks (13) umfasst, dass in einem oder mehreren oder allen Trainingsschritten des Trainings einige Konfigurationsparameter des Schülernetzwerks (13) konstant gehalten werden.

10. Verfahren nach einem der vorhergehenden Ansprüche, wobei jeweilige initiale Konfigurationsparameter des Schülernetzwerks (13) und des Diskriminatornetzwerks (15) jeweils mit Zufallszahlen oder

mit einem vorbestimmten, von dem Lehrernetzwerk (12) unabhängigen Zahlenmuster initialisiert werden.

11. Verfahren nach einem der vorhergehenden Ansprüche, wobei als das Lehrernetzwerk (12) ein tiefes neuronales Netzwerk mit einer oder zwei oder mehr als zwei Zwischenschichten bereitgestellt wird.

12. Verfahren nach einem der vorhergehenden Ansprüche, wobei bei dem Schülernetzwerks (13) eine Schichtanzahl und/oder eine Neuronenanzahl und/oder eine Anzahl an Neuronenverbindungen in Abhängigkeit von einer vorbestimmten Prozessorhardware (P) angepasst wird.

13. Verfahren nach Anspruch 12, wobei das Anpassen dahingehend erfolgt, dass das Schülernetzwerk (13) im Unterschied zum Lehrernetzwerk (13) eine vorbestimmte Echtzeitbedingung erfüllt, wenn es durch die Prozessorhardware (P) betrieben wird.

14. Verfahren nach einem der vorhergehenden Ansprüche, wobei das Netzwerk (12), welches als Lehrernetzwerk (12) verwendet wird, vor der Verwendung als Lehrernetzwerk (12) für eine der folgenden Funktionen Mustererkennung, Bildverarbeitung, Objektklassifikation, Verkehrssituationserkennung oder Verkehrsobjekterkennung trainiert wird und diese Funktion nach dem Training mittels des Schülernetzwerk (13) durchgeführt wird.

15. Verfahren nach Anspruch 14, wobei das trainierte Schülernetzwerk (13) in einer Verkehrsbeobachtungseinrichtung für eine Verkehrssituationserkennung oder Verkehrsobjekterkennung betrieben wird, wobei die Verkehrsbeobachtungseinrichtung in einer Infrastrukturkomponenten einer Straßeninfrastruktur installiert ist.

16. Vorrichtung (10) mit einer Prozessoreinrichtung (11), wobei die Prozessoreinrichtung (11) dazu eingerichtet ist, ein Verfahren nach einem der vorhergehenden Ansprüche durchzuführen.

17. Speichermedium mit einem darin gespeicherten Programmcode, wobei der Programmcode dazu eingerichtet ist, bei Ausführen durch eine Prozessoreinrichtung (11) ein Verfahren nach einem der Ansprüche 1 bis 15 durchzuführen.

Es folgen 4 Seiten Zeichnungen

Anhängende Zeichnungen

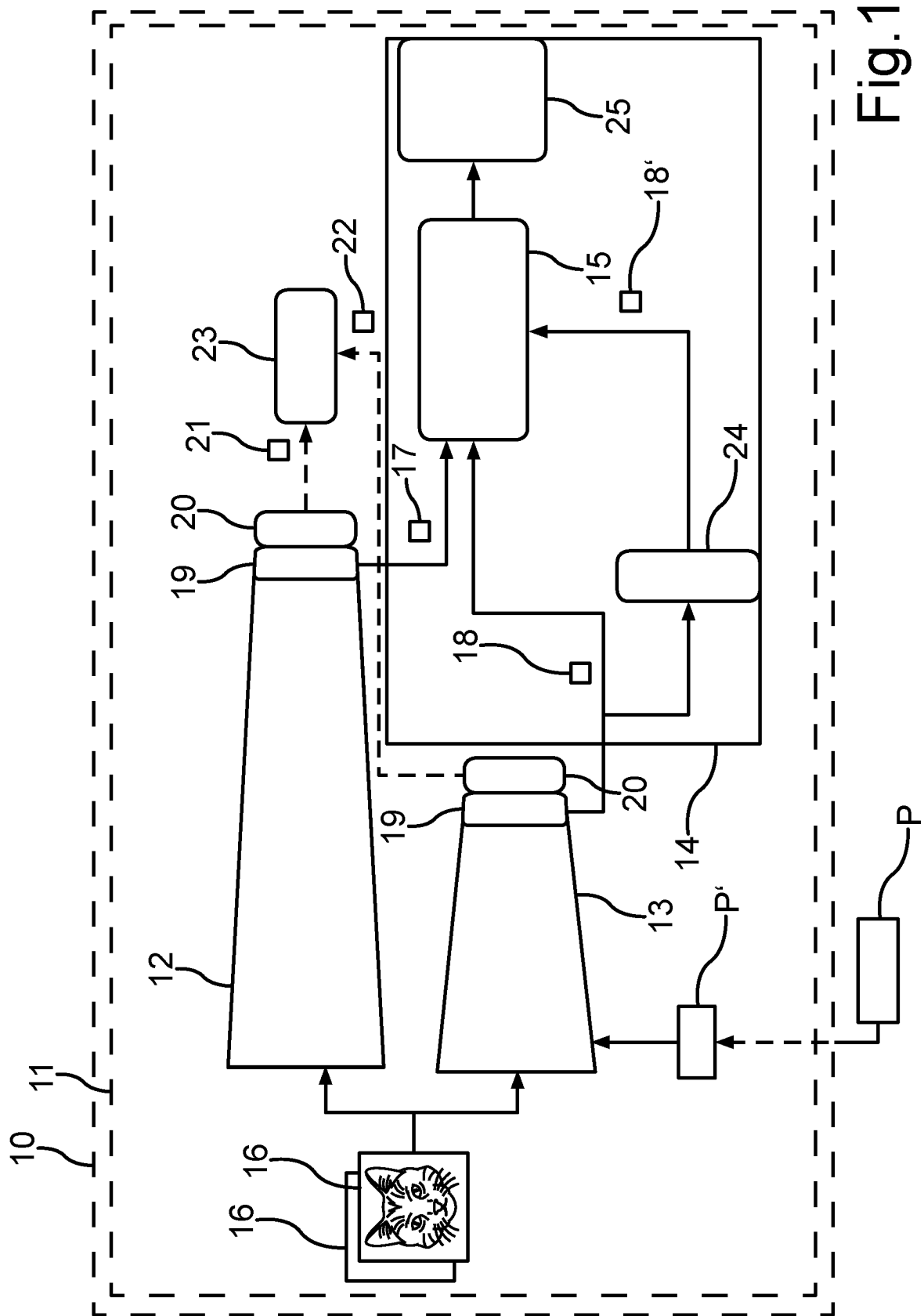


Fig. 1

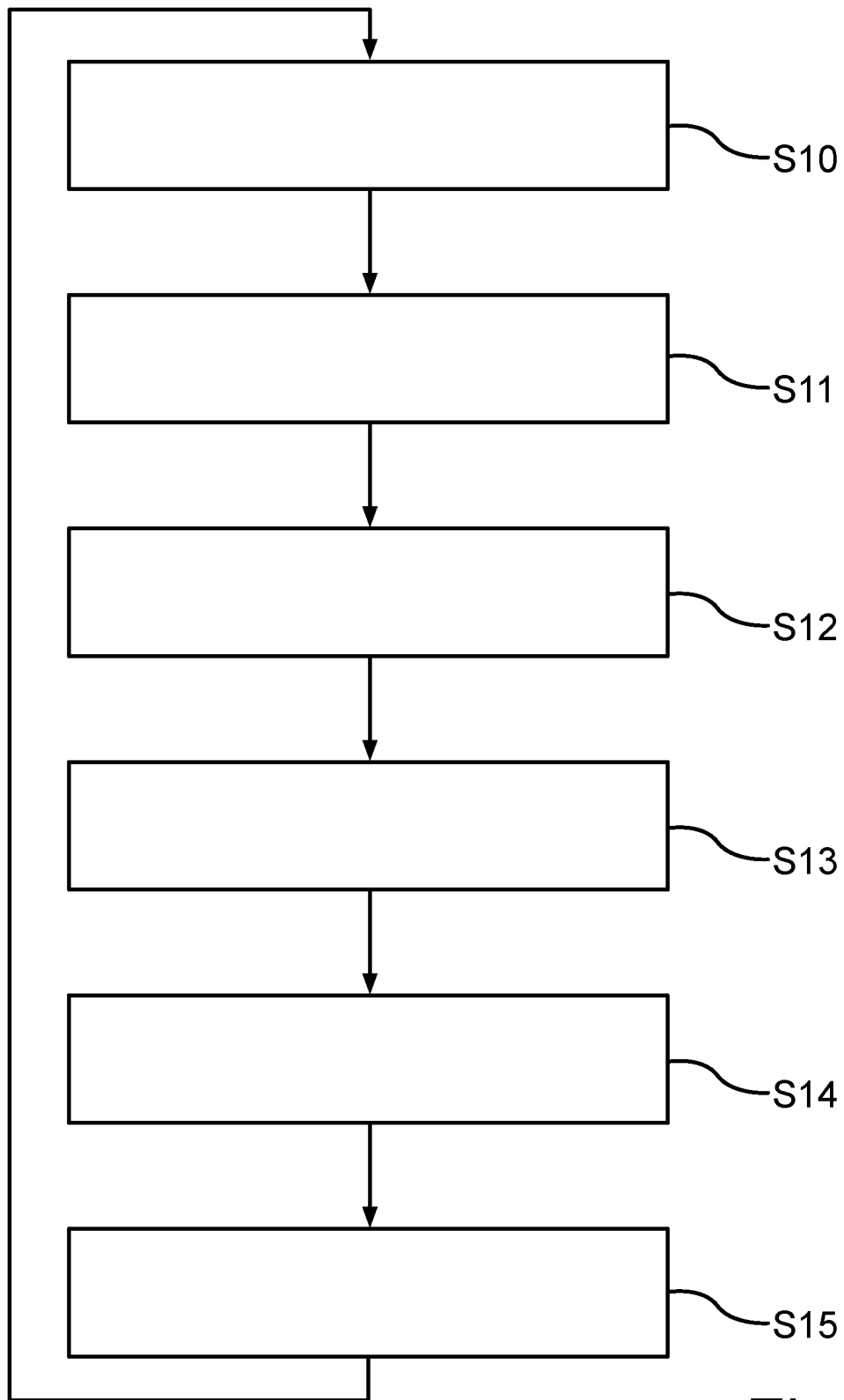


Fig.2

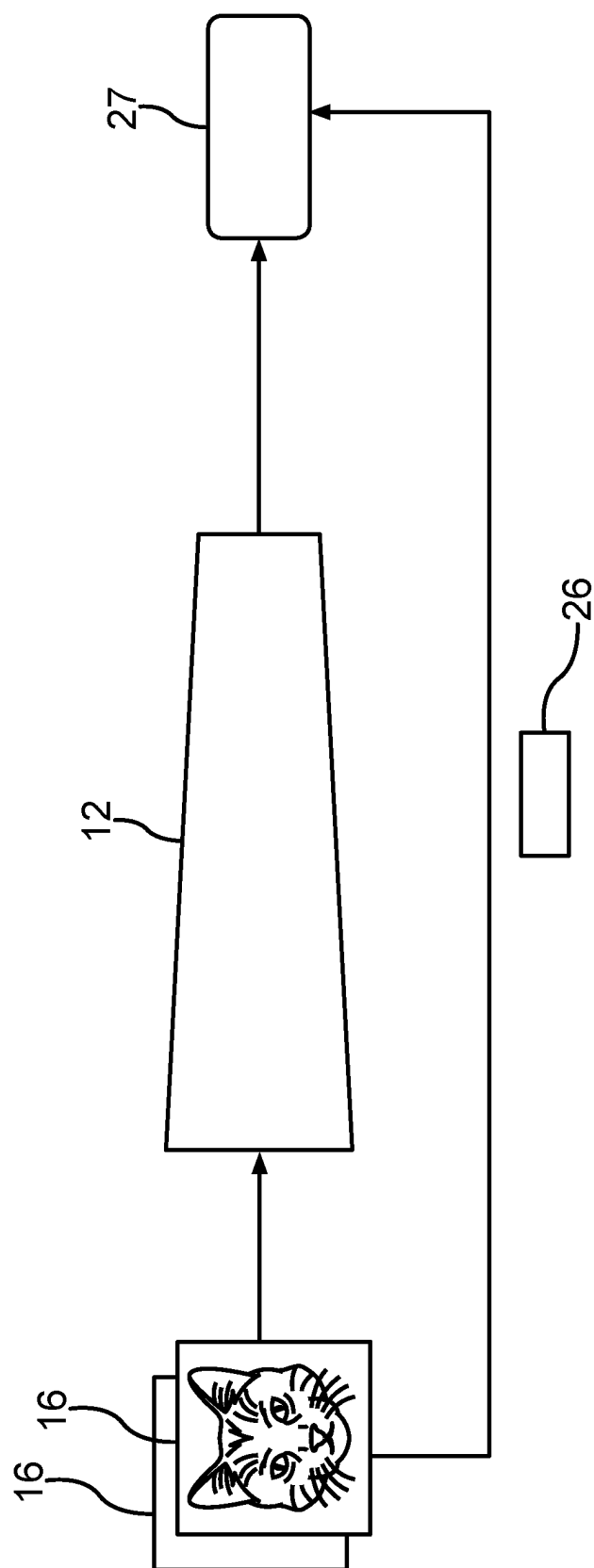


Fig.3

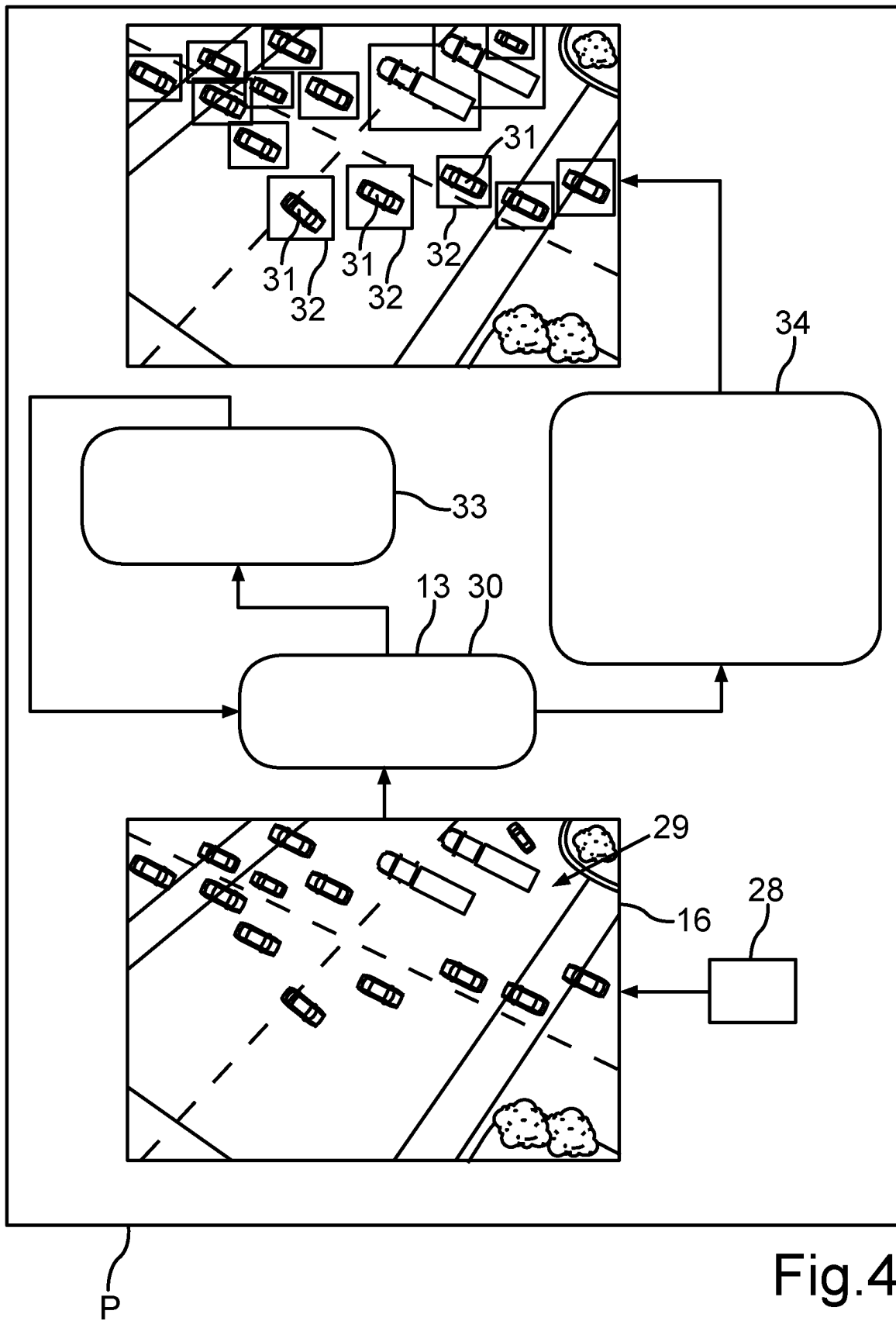


Fig.4