(19) Deutsches Patent- und Markenamt

(10) **DE 10 2018 206 806 A1** 2019.10.24

(12) # Offenlegungsschrift

(21) Aktenzeichen: **10 2018 206 806.2**
(22) Anmeldetag: **03.05.2018**
(43) Offenlegungstag: **24.10.2019**

(51) Int Cl.: ***G06N 3/08*** (2006.01)
G06K 9/62 (2006.01)

(66) Innere Priorität:
10 2018 206 199.8 23.04.2018

(71) Anmelder:
Robert Bosch GmbH, 70469 Stuttgart, DE

(72) Erfinder:
Hanselmann, Markus, 70563 Stuttgart, DE; Strauss, Thilo, 71636 Ludwigsburg, DE; Ulmer, Holger, 89077 Ulm, DE; Junginger, Andrej, 70197 Stuttgart, DE

(56) Ermittelter Stand der Technik:

HUANG, He; YU, Phillip S.; WANG, Changhu: An Introduction to Image Synthesis with Generative Adversarial Nets. In: CoRR, 12.03.2018, S. 1-17. https://arxiv.org/abs/1803.04469v1 [abgerufen am 24.01.2019]

Shrivastava, A. [et.al.]: Learning from Simulated and Unsupervised Images through Adversarial Training. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, S. 2242-2251. - ISSN 1063-6919. https://doi.org/10.1109/CVPR.2017.241 [abgerufen am 24.01.2019]

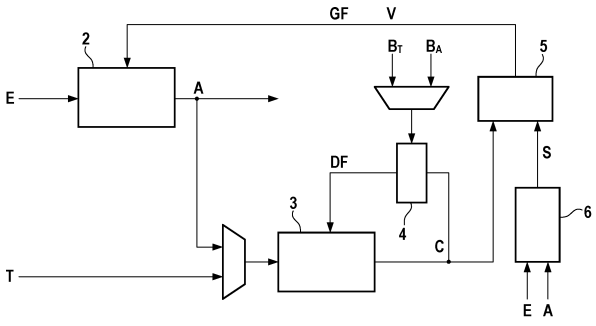
YOO, Donggeun [et.al.]: Pixel-Level Domain Transfer. In: CoRR, 28.11.2016, S. 1-17. https://arxiv.org/abs/1603.07442v3 [abgerufen am 24.01.2019]

Rechercheantrag gemäß § 43 PatG ist gestellt.

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen.

(54) Bezeichnung: **Verfahren und Vorrichtung zur Umsetzung eines Eingangsbildes einer ersten Domäne in ein Ausgangsbild einer zweiten Domäne**

(57) Zusammenfassung: Die Erfindung betrifft ein Verfahren zum Trainieren eines ersten neuronalen Netzes zur Umwandlung eines Eingangsbildes (E) einer ersten Domäne in ein Ausgangsbild (A) einer zweiten Domäne, wobei das Training auf für das Training bereitgestellten Eingangsbildern (E) der ersten Domäne und Trainingsbildern (T) der zweiten Domäne durchgeführt wird; mit folgenden Schritten:
- Bereitstellen eines GAN-Netzwerks mit einem Generatornetzwerk (2), das das erste neuronale Netz umfasst, und einem Diskriminatornetzwerk (3), das ein zweites neuronales Netz umfasst;
- Trainieren des Diskriminatornetzwerks (3) basierend auf einem Diskriminatorfehlerwert (DF) und einem oder mehreren Trainingsbildern (T) und/oder einem oder mehreren Ausgangsbildern (A), die durch Verarbeiten eines oder mehrerer der Eingangsbilder durch das Generatornetzwerk (2) erzeugt werden, wobei der Diskriminatorfehlerwert (DF) abhängig von einer jeweiligen Güte (C) des einen oder der mehreren Trainingsbilder (T) und/oder des einen oder der mehreren Ausgangsbilder bestimmt wird;
- Trainieren des Generatornetzwerks (2) basierend auf einem für das Training bereitgestellten Eingangsbild (E) und einem Generatorfehlerwert (GF), der von einer Güte (C) des von dem Generatornetzwerk (2) abhängig von dem Eingangsbild (E) bereitgestellten Ausgangsbilds (A) und einer Ähnlichkeitsgröße (S) zwischen dem Eingangsbild (E) und dem Ausgangsbild (A) abhängt, die ein Maß für eine strukturelle Ähnlichkeit angibt.



Beschreibung

Technisches Gebiet

[0001] Die Erfindung betrifft Verfahren zum Trainieren eines neuronalen Netzes zur Umsetzung eines Eingangsbildes einer ersten Domäne bzw. in einem ersten Darstellungsstil in ein Ausgangsbild einer zweiten Domäne bzw. in einem zweiten Darstellungsstil.

Technischer Hintergrund

[0002] Kraftfahrzeuge sind häufig mit Kamerasystemen ausgestattet, die eine Bildinformation über eine Fahrzeugumgebung, insbesondere ein Bild einer in Fahrtrichtung vorausliegenden Fahrzeugumgebung, erfassen. Diese Bildinformation wird zur Durchführung von Fahrerassistenzfunktionen zur Unterstützung des Fahrers und von autonomen Fahrfunktionen verwendet. Beispiele für solche Fahrerassistenzfunktionen können ein Erkennungssystem für Verkehrsschilder oder einen Bremsassistenten umfassen, der beispielsweise erkennt, dass sich ein Fußgänger in einem Kollisionsbereich vor dem Kraftfahrzeug befindet oder sich in diesen hineinbewegt.

[0003] Ein Problem bei der Entwicklung derartigen Funktionen besteht darin, dass nur unzureichend Bilddaten vorhanden sind, an denen diese Funktionen getestet bzw. trainiert werden können. Insbesondere ist es mühsam, Bilddaten für kritische Situationen zur Verfügung zu stellen.

[0004] Weiterhin beinhalten bereitgestellte Bilddaten in der Regel keine Metainformationen, die z. B. Bildsegmentierungsinformation aufweisen, d. h. die angeben, welche Pixelregionen der Bilddaten zu einem Fußgänger, zu einem Umgebungsbereich, zu einem Straßenbereich, zu einem Gebäudebereich und dergleichen gehören. Oftmals müssen derartige Bildinformationen manuell erstellt werden, was ein kostspieliger und vor allem zeitaufwendiger Prozess ist.

[0005] Bekannte Ansätze, Bilddaten für mögliche Verkehrssituationen als künstliches Kamerabild künstlich zu erzeugen, bestehen darin, gewünschte Verkehrssituationen per Skript, d. h. mit einer Formalsprache zu beschreiben und mit einer Graphik-Engine zu visualisieren. Die so ermittelten Bilder bzw. Bilddaten stellen die Verkehrssituationen jedoch künstlich und nicht photorealistisch dar, was für das Entwickeln und Testen von Fahrerassistenzfunktionen und autonomen Fahrfunktionen unter realistischen Bedingungen ungeeignet ist.

[0006] Weitere Verfahren sind aus dem Stand der Technik bekannt, die einen Stiltransfer von einem Eingangsbild zu einem Ausgangsbild vorschlagen. Während einfache Ansätze zum Training eines solchen Systems einander zugeordnete Bilddaten des Eingangsbilds und des Ausgangsbilds verwenden, die beide denselben Bildinhalt aufzeigen und nur im Stil (mit ihrer Domäne) voneinander abweichen, können fortgeschrittene Verfahren Eingangs- und Ausgangsbilddaten der entsprechenden Stile verwenden, die keinen Bezug zueinander aufweisen müssen.

[0007] Ein Nachteil der oben beschriebenen Verfahren besteht darin, dass während des Trainings eine sogenannte Cycle Consistency berechnet werden muss, wodurch im Training explizit die Eingangsbilddaten in die Ausgangsbilddaten und umgekehrt berechnet werden müssen, was das Training sehr rechenintensiv und damit zeitaufwendig macht.

Offenbarung der Erfindung

[0008] Erfindungsgemäß sind ein Verfahren zum Trainieren eines neuronalen Netzes zur Umsetzung eines Eingangsbildes einer ersten Domäne in ein Ausgangsbild einer zweiten Domäne gemäß Anspruch 1 sowie eine entsprechende Vorrichtung gemäß dem nebengeordneten Anspruch vorgesehen.

[0009] Weitere Ausgestaltungen sind in den abhängigen Ansprüchen angegeben.

[0010] Gemäß einem ersten Aspekt ist ein Verfahren zum Trainieren eines ersten neuronalen Netzes zur Umwandlung eines Eingangsbildes einer ersten Domäne in ein Ausgangsbild einer zweiten Domäne vorgesehen, wobei das Training auf für das Training bereitgestellten Eingangsbildern der ersten Domäne und Trainingsbildern der zweiten Domäne durchgeführt wird; mit folgenden Schritten:

- Bereitstellen eines GAN-Netzwerks mit einem Generatornetzwerk, das das erste neuronale Netz umfasst, und einem Diskriminatornetzwerk, das ein zweites neuronales Netz umfasst;

- Trainieren des Diskriminatornetzwerks basierend auf einem Diskriminatorfehlerwert und einem oder mehreren Trainingsbildern und/oder einem oder mehreren Ausgangsbildern, die durch Verarbeiten eines oder mehrerer der Eingangsbilder durch das Generatornetzwerk erzeugt werden, wobei der Diskriminatorfehlerwert abhängig von einer jeweiligen Güte des einen oder der mehreren Trainingsbilder und/oder des einen oder der mehreren Ausgangsbilder bestimmt wird;

- Trainieren des Generatornetzwerks basierend auf einem für das Training bereitgestellten Eingangsbild und einem Generatorfehlerwert, der von einer Güte des von dem Generatornetzwerk abhängig von dem Eingangsbild bereitgestellten

Ausgangsbilds und einer Ähnlichkeitsgröße zwischen dem Eingangsbild und dem Ausgangsbild abhängt, die ein Maß für eine strukturelle Ähnlichkeit angibt.

[0011] Ziel des obigen Verfahrens ist es, ein neuronales Netz so zu trainieren, dass ein vorgegebenes Eingangsbild in ein Ausgangsbild umgewandelt wird. Dabei sollen Eingangs- und Ausgangsbild unterschiedliche Stile aufweisen, d. h. die Eingangsbilddaten sollen in einer ersten Domäne und die Ausgangsbilddaten in einer zweiten Domäne zur Verfügung stehen. Die Stile entsprechen Darstellungsstilen, wie z.B. eine Segmentierungsdarstellung, in der z.B. farblich unterschiedliche Bereiche unterschiedlichen Objekten oder Bildbereichen zugeordnet sind, ein photorealistisches Bild, ein Comic-Bild, eine Strichzeichnung, eine Aquarell-Skizze und dergleichen.

[0012] Zum Erstellen und Testen von Fahrerassistenzfunktionen und/oder autonomen Fahrfunktionen für ein Kraftfahrzeug, die auf einer Auswertung von Kamerabildern der aktuellen Fahrzeugumgebung basieren, ist es notwendig, eine ausreichende Anzahl von photorealistischen Bildern der Fahrzeugumgebung bereitzustellen. Diese Bilder sollen Kamerabilder ersetzen und von solchen möglichst nicht unterscheidbar sein. Diese Bilder können zudem optional mit einer Metainformation zur Verfügung gestellt werden, die beispielsweise eine Segmentierungsinformation umfassen, die Bildbereiche des photorealistischen Bildes bestimmten Objekten oder Hintergründen zuordnet. Somit ist eine wichtige Anwendung für ein so trainiertes neuronales Netz die Umwandlung eines z. B. durch eine Skriptsprache oder als Handskizze beschriebenen Eingangsbilds in ein künstlich erzeugtes photorealistisches Ausgangsbild, das dem Eingangsbild inhaltlich bzw. szenarisch entspricht, jedoch in der Darstellungsweise, dem Darstellungsstil, von diesem abweicht.

[0013] Im Folgenden wird von einer Umwandlung eines Eingangsbilds eines ersten Stiles in ein Ausgangsbild eines zweiten Stiles bzw. eines Eingangsbilds in einer ersten Domäne in ein Ausgangsbild in einer zweiten Domäne gesprochen, um diesen Generierungsprozess zu beschreiben. Beispielsweise kann ein Eingangsbild, das lediglich Bildbereiche für bestimmte Objekte und/oder Hintergründe angibt, wie beispielsweise Bildbereiche zur Darstellung einer Person, eines Fahrradfahrers, eines Straßenbereichs, eines Bebauungsbereichs, eines Vegetationsbereichs und dergleichen, so durch das trainierte neuronale Netz verarbeitet werden, dass die entsprechenden Bildbereiche mit realistischen Strukturen der entsprechenden Objekte versehen werden.

[0014] Dazu sieht das obige Verfahren vor, ein GAN-Netzwerk zu verwenden (GAN: Generative Adversarial Network), bei dem ein Generatornetzwerk, das einem ersten neuronalen Netz entspricht, mithilfe eines Diskriminatornetzwerks, das einem zweiten neuronalen Netz entspricht, trainiert werden soll. Das Generatornetzwerk erzeugt dann aus bereitgestellten Eingangsbilddaten in einer ersten Domäne Ausgangsbilddaten in einer zweiten Domäne.

[0015] Bei einem GAN-Netzwerk wird die Güte des Trainings des Generatornetzwerks mithilfe des Diskriminatornetzwerks verbessert. Das Diskriminatornetzwerk liefert für das Training des Generatornetzwerks als relevante Information ein Bewertungslabel für das vom Generatornetzwerk generierte Ausgangsbild. Zum Bereitstellen des Bewertungslabels wird das Diskriminatornetzwerk so trainiert, dass dieses bewerten kann, ob ein an dessen Eingang bereitgestelltes Bild ein Bild in einer zweiten Domäne ist. Das Diskriminatornetzwerk wird zeitgleich oder im Wechsel mit dem Generatornetzwerk basierend auf vom Generatornetzwerk generierten Ausgangsbildern und Trainingsbildern in einer zweiten Domäne trainiert, wobei den Trainingsbildern ein Bewertungslabel zugeordnet wird, das ein hohes Zuordnungsmaß zur zweiten Domäne angibt (d. h. angibt, dass die betreffenden Bilder der zweiten Domäne zuzuordnen sind). Zudem werden dem Diskriminatornetzwerk die Ausgangsbilder zugeführt, die vom Generatornetzwerk erstellt wurden, zusammen mit einem Bewertungslabel, das ein niedriges Zuordnungsmaß zur zweiten Domäne angibt (d. h. angibt, dass die betreffenden Bilder der zweiten Domäne künstlich durch das Generatornetzwerk erzeugt wurden).

[0016] Generatornetzwerk und Diskriminatornetzwerk können wechselweise trainiert werden, wodurch iterativ beide neuronale Netze verbessert werden und das Generatornetzwerk letztlich lernt, ein bereitgestelltes Eingangsbild in der ersten Domäne in ein Ausgangsbild in der zweiten Domäne umzuwandeln.

[0017] Zum Training des Generatornetzwerks und des Diskriminatornetzwerks werden Verlustfunktionen bzw. Kostenfunktionen verwendet. Zum Training des Generatornetzwerks wird als Kostenfunktion eine Generatorfehlerfunktion verwendet, die zwei Teile umfasst. Ein erster Teil erzwingt, dass das generierte Ausgangsbild der zweiten Domäne zugeordnet wird. Hierfür wird das durch das Generatornetzwerk generierte Ausgangsbild dem Diskriminatornetzwerk zugeführt und der Abstand zum gewünschten Bewertungslabel (Bewertungslabel für ein Trainingsbild der zweiten Domäne) minimiert. Der zweite Teil stellt sicher, dass die Bildinhalte des durch das Generatornetzwerk generierten Ausgangsbildes dem Ursprungsbild entsprechen, indem ein struktureller Abstand des Ausgangsbildes zum Eingangsbild minimiert wird, d. h. das Ausgangsbild unterscheidet sich von dem Eingangsbild lediglich durch den Darstel-

lungsstil (Domäne) aber nur wenig durch den Bildinhalt bzw. die dargestellte Szene.

[0018] Der strukturelle Abstand kann beispielsweise durch einen Ähnlichkeitswert bestimmt werden, der ein Maß für die strukturelle Ähnlichkeit von zwei Bildern in unterschiedlichen Domänen ist. Beispielsweise eignet sich hierfür ein SSIM-Index (SSIM: Structural Similarity Index), der die strukturelle Ähnlichkeit zwischen dem Eingangs- und dem Ausgangsbild in bekannter Weise angibt.

[0019] Auf diese Weise wird ermöglicht, dass das Generatornetzwerk darauf trainiert wird, ein Eingangsbild in der ersten Domäne bzw. eines ersten Darstellungsstils in ein Ausgangsbild in einer zweiten Domäne, d.h. eines zweiten Darstellungsstils, umzuwandeln. Dazu müssen die Eingangsbilder des ersten Darstellungsstils und die Trainingsbilder des zweiten Darstellungsstils vorgegeben werden, wobei eine Ähnlichkeit oder Identität der Darstellung der Eingangsbilder und der Trainingsbilder nicht notwendig ist, d. h. es ist nicht notwendig, Eingangsbilder bereitzustellen, die sich von den Trainingsbildern nur durch den Darstellungsstil unterscheiden.

[0020] Durch das obige Verfahren kann somit ein neuronales Netz (Generatornetzwerk) trainiert werden, das automatisiert und überwacht aus synthetischen Eingangsbildern, die beispielsweise eine Verkehrssituation schematisch bzw. stilisiert zeigen, photorealistische Ausgangsbilder der entsprechenden Verkehrssituation generiert. Die Ausgangsbilder können dann dazu verwendet werden, Fahrerassistenzfunktionen bzw. autonome Fahrfunktionen zu entwickeln und/oder zu testen. Ein Vorteil besteht insbesondere darin, dass Situationen erstellt werden können, die in der Realität nicht getestet werden können, wie z. B. eine auf die Fahrbahn rennende Person zum Testen eines Bremsassistenzsystems bzw. zum Testen eines Ausweichverhaltens einer autonomen Fahrfunktion.

[0021] Insgesamt lässt sich durch das oben beschriebene Trainingsverfahren eine deutlich verbesserte Umsetzung eines Eingangsbildes eines ersten Darstellungsstils in ein entsprechendes Ausgangsbild eines zweiten Darstellungsstils erreichen, wobei das Trainingsverfahren in einfacher Weise implementiert werden kann und eine hohe Zuverlässigkeit und Robustheit aufweist. Auch führt das obige Trainingsverfahren zu besseren Resultaten, d. h. eine verbesserte präzisere Umsetzung des Eingangsbildes des ersten Darstellungsstils in das Ausgangsbild des zweiten Darstellungsstils, als entsprechende herkömmliche Verfahren.

[0022] Weiterhin kann das Trainieren des Diskriminatornetzwerks und des Generatornetzwerks gleichzeitig oder wechselweise wiederholt durchgeführt werden, insbesondere mithilfe eines Backpropagation-Verfahrens, bis eine Abbruchbedingung erfüllt ist.

[0023] Es kann vorgesehen sein, dass die Abbruchbedingung erfüllt ist, wenn eine Anzahl von Durchgängen oder eine vorbestimmte Güte der vom Generatornetzwerk generierten Ausgangsbilder erreicht ist.

[0024] Weiterhin kann die Güte des einen oder der mehreren Trainingsbilder und/oder des einen oder der mehreren Ausgangsbilder jeweils durch das Diskriminatornetzwerk bestimmt werden und einer Bewertung entsprechen, in welchem Maß es sich bei dem betreffenden Bild um ein Bild der zweiten Domäne handelt.

[0025] Insbesondere kann der Diskriminatorfehlerwert abhängig von einem Abweichungsmaß für die Abweichung zwischen der jeweiligen Güte des betreffenden einen Trainingsbildes oder der betreffenden mehreren Trainingsbilder und einem Bewertungslabel, das ein Trainingsbild als ein echtes Bild der zweiten Domäne angibt, und abhängig von einem Abweichungsmaß für die Abweichung zwischen der jeweiligen Güte des betreffenden einen Ausgangsbildes oder der betreffenden mehreren Ausgangsbilder und einem Bewertungslabel, das ein von dem Generatornetzwerk generiertes Ausgangsbild als unechtes Bild der zweiten Domäne angibt, bestimmt werden, wobei das Abweichungsmaß insbesondere einem Mean Squared Error oder einer binären Cross Entropy entspricht.

[0026] Es kann vorgesehen sein, dass die Ähnlichkeitsgröße von einem SSIM-Index für eine strukturelle Ähnlichkeit zwischen einem der Eingangsbilder und einem durch das Generatornetzwerk aus dem betreffenden Eingangsbild generierten Ausgangsbild abhängt oder diesem entspricht.

[0027] Weiterhin kann das erste und/oder das zweite neuronale Netz als Convolutional Neural Network) (faltendes neuronales Netz) ausgebildet sein, wobei insbesondere das erste und/oder das zweite neuronale Netz eine Hintereinanderschaltung von einigen Convolutional Layer-Blöcken (Faltungsblöcken), einigen ResNet-Blöcken und einigen Deconvolutional-Blöcken aufweisen, wobei jeder der Blöcke als eine Aktivierungsfunktion eine ReLU-, leaky-ReLU-, tanh- oder Sigmoid-Funktion enthalten kann.

[0028] Weiterhin kann der Generatorfehlerwert von einem Abweichungsmaß für die Abweichung zwischen der jeweiligen Güte des von dem Generatornetzwerk abhängig von dem Eingangsbild bereitgestellten Ausgangsbilds und einem Bewertungslabel aus dem Diskriminatornetzwerk, das ein Bild der zweiten Domäne angibt, abhängen, wobei das Ab-

weichungsmaß insbesondere einem Mean Squared Error oder einer binären Cross Entropy entspricht.

[0029] Gemäß einer Ausführungsform kann das Training des Diskriminatornetzwerks und/oder des Generatornetzwerks nur dann durchgeführt werden, wenn eine von dem aktuellen Diskriminatorfehlerwert und/oder von dem Generatorfehlerwert abhängige Bedingung erfüllt ist.

[0030] Weiterhin kann ein Verfahren zum Bereitstellen einer Steuerung für ein technisches System, insbesondere für einen Roboter, ein Fahrzeug, ein Werkzeug oder eine Werkmaschine, vorgesehen sein, wobei das obige Verfahren zum Trainieren eines ersten neuronalen Netzes ausgeführt wird, wobei das trainierte erste neuronale Netz verwendet wird, um Trainingsbilder, d.h. Ausgangsbilder der zweiten Domäne, zu erzeugen, mit denen die Steuerung, die insbesondere ein neuronales Netz enthält, trainiert wird. Insbesondere kann das technische System mithilfe der Steuerung betrieben werden.

[0031] Gemäß einem weiteren Aspekt ist eine Verwendung eines ersten neuronalen Netzes, das entsprechend dem obigen Verfahren trainiert ist, zum Generieren von photorealistischen Ausgangsbildern in einer zweiten Domäne abhängig von vorgegebenen Eingangsbildern in einer ersten Domäne, die insbesondere über eine Skript-basierte Beschreibung erstellt werden

[0032] Weiterhin können die erzeugten photorealistischen Ausgangsbilder als künstliche Kamerabilder zum Herstellen eines Klassifikators für Umgebungssituationen verwendet werden.

[0033] Gemäß einem weiteren Aspekt ist ein GAN-Netzwerk zum Trainieren eines ersten neuronalen Netzes zur Umwandlung eines Eingangsbildes einer ersten Domäne in ein Ausgangsbild einer zweiten Domäne, wobei das Training auf für das Training bereitgestellten Eingangsbildern der ersten Domäne und Trainingsbildern der zweiten Domäne durchgeführt wird, wobei das GAN-Netzwerk ein Generatornetzwerk, das das erste neuronale Netz umfasst, und ein Diskriminatornetzwerk aufweist, das ein zweites neuronales Netz umfasst, wobei das GAN-Netzwerk ausgebildet ist, um

- das Diskriminatornetzwerk basierend auf einem Diskriminatorfehlerwert und einem oder mehreren Trainingsbildern und/oder einem oder mehreren Ausgangsbildern, die durch Verarbeiten eines oder mehrerer der Eingangsbilder durch das Generatornetzwerk erzeugt werden, zu trainieren, wobei der Diskriminatorfehlerwert abhängig von einer jeweiligen Güte des einen oder der mehreren Trainingsbilder und/oder des einen oder der mehreren Ausgangsbilder bestimmt wird; und
- das Generatornetzwerk basierend auf einem für das Training bereitgestellten Eingangsbild und einem Generatorfehlerwert zu trainieren, der von einer Güte des von dem Generatornetzwerk abhängig von dem Eingangsbild bereitgestellten Ausgangsbilds und einer Ähnlichkeitsgröße zwischen dem Eingangsbild und dem Ausgangsbild abhängt, die ein Maß für eine strukturelle Ähnlichkeit angibt.

Figurenliste

[0034] Ausführungsformen werden nachfolgend anhand der beigefügten Zeichnungen näher erläutert. Es zeigen:

Fig. 1a und **Fig. 1b** beispielhafte Darstellungen eines Bildes eines ersten Darstellungsstils und eines zugeordneten Bildes eines zweiten Darstellungsstils;

Fig. 2 ein Blockdiagramm zur Veranschaulichung eines Systems zum Trainieren eines GAN-Netzes zur Umsetzung eines Eingangsbildes eines ersten Darstellungsstils und eines Ausgangsbildes eines zweiten Darstellungsstils; und

Fig. 3 ein Flussdiagramm zur Veranschaulichung eines Verfahrens zum Trainieren eines neuronalen Netzes zur Umsetzung eines Eingangsbildes in ein Ausgangsbild eines davon verschiedenen Darstellungsstils.

Beschreibung von Ausführungsformen

[0035] Es soll ein neuronales Netz trainiert werden, das in der Lage ist, ein Eingangsbild in ein Ausgangsbild umzuwandeln. Ziel ist es, dass das Eingangsbild in einer ersten Domäne, d. h. in einem ersten Darstellungsstil, bereitgestellt wird und in ein dem Eingangsbild entsprechendes Ausgangsbild in einer zweiten Domäne, d. h. in einem zweiten von dem ersten verschiedenen Darstellungsstil, bereitgestellt wird. „Darstellungsstil“ bezeichnet hierin eine Darstellungsweise einer in dem entsprechenden Bild enthaltenen Information.

[0036] So können beispielsweise ein Segmentierungsbild, das eine Segmentierung von Objekt- und Hintergrundbereichen eines photorealistischen Bilds angibt, oder ein sonstiges künstlich erzeugtes (synthetisches Bild), wie z.B. eine Skizze, als Eingangsbild eine Vorlage darstellen, aus der ein photorealistisches Bild als Ausgangsbild generiert wird, so dass das Eingangsbild und das Ausgangsbild unterschiedlichen Darstellungsstilen entsprechen. **Fig. 1a** und **Fig. 1b** zeigen beispielhafte Darstellungen eines synthetischen Bildes bzw. eines dem synthetischen Bild entsprechenden photorealistischen Bildes in Skizzenform und als realistische Darstellungen.

[0037] Eine mögliche Anwendung eines so trainierten neuronalen Netzes könnte darin bestehen, ein vorgegebenes Eingangsbild in Form eines Segmentierungsbildes, bei dem lediglich Segmentierungsbereiche vorgegeben sind, in ein künstlich erzeugtes photorealistisches Ausgangsbild umzuwandeln. So kann, wie in **Fig. 1** als Realbild und als Skizzenbild gezeigt, beispielsweise ein Segmentierungsbild (**Fig. 1a)**, in dem lediglich Flächen gekennzeichnet sind, die beispielsweise Darstellungsbereiche für einen Fahrbahnbereich, einen Bebauungsbereich, einen Vegetationsbereich, von Fremdfahrzeugen, von Fußgängern, von Fahrradfahrern oder von sonstigen Objekten aufweisen, in ein entsprechendes photorealistisches Bild (**Fig. 1b)** umgewandelt werden. Ein solches photorealistisches Bild kann dann in einer Test- oder Entwicklungsumgebung zum Testen und/oder Erstellen von Fahrerassistenzfunktionen oder autonomen Fahrfunktionen verwendet werden.

[0038] Zum Trainieren eines neuronalen Netzes kann ein System verwendet werden, das strukturell dem Blockschaltbild der **Fig. 2** entspricht. **Fig. 2** zeigt im Wesentlichen eine Grundstruktur eines GAN-Netzwerks **1** mit einem Generatornetzwerk **2**, das ein erstes neuronales Netz umfasst, und einem Diskriminatornetzwerk **3**, das ein zweites neuronales Netz umfasst. Das erste und/oder zweite neuronale Netz können insbesondere als faltende neuronale Netze (convolutional neural networks) oder andere Arten von neuronalen Netzen ausgebildet sein.

[0039] Für das erste neuronale Netz des Generatornetzwerks **2** sind verschiedene an sich bekannte Architekturen denkbar. Insbesondere kann eine Hintereinanderschaltung von einigen Convolutional Layer-Blöcken (Faltungsblöcken), einigen ResNet-Blöcken und einigen Deconvolutional-Blöcken gewählt werden. Jeder dieser Blöcke kann optional eine Batch- oder andere Art der Normalisierung umfassen. Jeder der Blöcke kann weiterhin keine, eine oder mehrere Aktivierungsfunktionen enthalten, wie beispielsweise eine ReLU-, leaky-ReLU-, tanh- oder Sigmoid-Funktion.

[0040] Für das zweite neuronale Netz des Diskriminatornetzwerks **3** können ebenfalls verschiedene an sich bekannte Netzwerkarchitekturen vorgesehen werden. Als Netzwerkarchitektur können eine Hintereinanderschaltung von Blöcken, wie mehreren Convolutional Layer-Blöcken (Faltungsblöcken), einigen ResNet-Blöcken und einigen Deconvolutional-Blöcken verwendet werden. Jeder dieser Blöcke kann ein Batch- oder eine andere Art der Normalisierung enthalten. Weiterhin kann jeder der Blöcke keine, eine oder mehrere Aktivierungsfunktionen enthalten, wie beispielsweise eine ReLU-, leaky-ReLU-, tanh- oder Sigmoid-Funktion.

[0041] Das Generatornetzwerk **2** ist ausgebildet, um basierend auf einem Eingangsbild **E** eines ersten Darstellungsstils ein Ausgangsbild **A** eines zweiten Darstellungsstils zu erzeugen. Das Eingangsbild **E** kann ein Bild mit einem oder mehreren Farbkanälen, insbesondere drei Farbkanälen, sein und das Ausgangsbild **A** ein Tensor desselben oder abweichenden Formats. Alternativ kann dem Eingangsbild **E** ein Zufallstensor hinzugefügt werden, der dafür sorgt, dass das Ausgangsbild **A** eine höhere Variabilität aufweist.

[0042] Das Generatornetzwerk **2** wird basierend auf einem bereitgestellten Generatorfehlerwert **GF** trainiert, insbesondere mithilfe eines Backpropagation-Verfahrens. Der Generatorfehlerwert **GF** wird in einem Bewertungsblock **4** erzeugt, der einerseits die strukturelle Ähnlichkeit **S** bzw. Unähnlichkeit des Eingangsbilds **E** und des von dem Generatornetzwerk **2** basierend auf einem vorgegebenen Eingangsbild **E** generierten Ausgangsbilds **A** (Bildähnlichkeit (Ähnlichkeit des Bildinhalts bzw. der Szene) ungeachtet der Domäne bzw. des Darstellungsstils) und andererseits die Güte **C** des Ausgangsbildes **A** angibt. Die Güte **C** des Ausgangsbildes **A** gibt die Nähe des Darstellungsstils des Ausgangsbildes **A** zu dem Darstellungsstil von vorgegebenen Trainingsbildern **T** an.

[0043] Die Güte **C** des Ausgangsbildes **A** wird mithilfe des Diskriminatornetzwerks **3** bestimmt, dem als Eingang das erstellte Ausgangsbild **A** bereitgestellt wird. Durch Berücksichtigung der Güte **C** beim Training des Generatornetzwerks **3** wird erreicht, dass das generierte Ausgangsbild **A** den zweiten Darstellungsstil annimmt. Zusätzlich wird durch Berücksichtigung der strukturellen Ähnlichkeit **S** zwischen Eingangsbild E und Ausgangsbild **A** erreicht, dass die Bilder denselben Bildinhalt haben.

[0044] Zum Training können dem Diskriminatornetzwerk **3** Trainingsbilder **T** zugeführt werden, die Bilder des zweiten Darstellungsstils sind und die jeweils mit einem Bewertungslabel $\mathbf{B_T}$ versehen sind, das den zweiten Darstellungsstil der Trainingsbilder bestätigt. Beispielsweise können die Trainingsbilder **T** mit einem Bewertungslabel $\mathbf{B_T}$ von 1 versehen werden, was angibt, dass die Trainingsbilder T dem zweiten Darstellungsstil entsprechen. Um die Diskriminierungsfähigkeit des Diskriminatornetzwerks **3** zu verbessern, können zum Training dem Diskriminatornetzwerk **3** auch die durch das Generatornetzwerk **2** erzeugten Ausgangsbilder A bereitgestellt, die mit einem Bewertungslabel $\mathbf{B_A}$ von 0 versehen sind, was angibt, dass der Darstellungsstil dieser Bilder sich von dem zweiten Darstellungsstil erheblich unterscheidet. Durch Bereitstellen der Trainingsbilder T und der Ausgangsbilder mit den zugehörigen Bewertungslabels $\mathbf{B_T}$, $\mathbf{B_A}$ kann das Diskriminatornetzwerk **3** z. B. mithilfe des Backpropagation-Verfahrens oder eines sonstigen Trainingsverfahrens trainiert werden,

die Güte **C** von durch das Generatornetzwerk **2** bereitgestellten Ausgangsbildern **A** zu bestimmen.

[0045] Beim Training des Diskriminatornetzwerks **3** kann dieses mithilfe einer Diskriminatorfehlerfunktion DFK, wie z. B. einer Mean Squared Error, binären Cross Entropy oder anderer geeigneter Kostenfunktionen trainiert werden. Dadurch erhält das Diskriminatornetzwerk **3** durch Beeinflussung des Generatorfehlerwerts die Fähigkeit, dass das Generatornetzwerk **2** nicht nur Ausgangsbilder **A** erzeugt, die dem zweiten Darstellungsstil entsprechen, sondern dass gleichzeitig die Ausgangsbilder **A** denselben Bildinhalt haben, wie das dem Generatornetzwerk **2** zugeführte Eingangsbild E des ersten Darstellungsstils.

[0046] Durch eine wechselseitige oder gleichzeitige Trainingsphase von Generatornetzwerk **2** und Diskriminatornetzwerk **3** können diese iterativ verbessert werden. Dabei wird das Generatornetzwerk **2** mit dem Generatorfehlerwert **GF** mithilfe eines Backpropagation-Verfahrens oder eines sonstigen Trainingsverfahren trainiert, wobei der Generatorfehlerwert **GF** durch die strukturelle Ähnlichkeit zwischen dem Eingangsbild **E** und dem durch das Generatornetzwerk **2** generierten Ausgangsbild **A** und durch die von dem Diskriminatornetzwerk **3** bestimmte Güte **C** des durch das Generatornetzwerk **2** generierten Ausgangsbilds A bestimmt ist.

[0047] Abhängig von dem Trainingsbild T als Eingang des Diskriminatornetzwerks **3** wird ein Tensor $\mathbf{B_x}$ bereitgestellt. Dieser kann mehrdimensional sein oder einer reellen Zahl entsprechen. Der Tensor $\mathbf{B_x}$ entspricht dem Bewertungslabel und kann für die Trainingsbilder elementweise jeweils 1 und für die durch das Generatornetzwerk erzeugten Bilder jeweils 0 angeben. Die Bewertungslabel entsprechen somit B_1 für ein Trainingsbild **T** und $\mathbf{B_0}$ für ein von dem Generatornetzwerk generiertes Ausgangsbild **A**. Die Dimension des Bewertungslabels **B** ist im Wesentlichen frei wählbar und abhängig von der gewählten Netzwerkarchitektur. Das Bewertungslabel **B** kann auch mit einer anderen Normierung versehen sein, und insbesondere können sogenannte weiche Bewertungslabel B angesetzt werden. d. h., dass anstelle der Werte **1** und **0** auch entsprechend leicht verrauschte Werte angenommen werden können, wodurch die Stabilität des Trainings je nach Anwendungsfall verbessert werden kann.

[0048] Die Abbildung des Diskriminatornetzwerks **3** entspricht $D_{\theta\,D}$, wobei θ_D die zu optimierenden Diskriminatorparameter (Gewichtungen) des neuronalen Netzes des Diskriminatornetzwerks sind. Analog entspricht die Abbildung, die vom Generatornetzwerk **2** durchgeführt wird, $G_{\theta\,C}$, wobei θ_G die zu optimierenden Generatorparameter (Gewichtungen) des neuronalen Netzes des Generatornetzwerks **2** sind.

[0049] Die Diskriminatorfehlerfunktion zum Training des Diskriminatornetzwerks **2** dient der Bestimmung des Diskriminatorfehlerwerts **DF**, der im Training zur Parameteroptimierung der Diskriminatorparameter $\mathbf{\theta_D}$ verwendet wird. Die Verlustfunktion weist mehrere Summanden $\mathbf{l_D}$ auf. Die Diskriminatorfehlerfunktion ist in dem Diskriminatorbewertungsblock **4** implementiert und bewertet zum Training eine Abweichung $\mathbf{l_D}$ zwischen einer für ein angelegtes Bild (Trainingsbild T oder Ausgangsbild A) ermittelten Güte C (T) = $D_{\theta\,D}(T)$ oder C (A) = $D_{\theta\,D}(A)$ und einem dem angelegten (zugeführten) Bild zugeordneten Bewertungslabel $\mathbf{B_A}$, $\mathbf{B_T}$ (z. B. 1 für Trainingsbild T oder 0 für ein Ausgangsbild A). Die Diskriminatorfehlerfunktion DFK, die für dieses Training des Diskriminatornetzwerks **3** verwendet wird, muss ein Abweichungsmaß $\mathbf{l_D}$ realisieren, wie weit C(T), C(A) und das entsprechende Bewertungslabel $\mathbf{B_a}$, , $\mathbf{B_T}$ voneinander abweichen. Zur Bestimmung des Abweichungsmaßes $\mathbf{l_D}$ kann man jede geeignete Funktion zur Abstandsbewertung annehmen, wobei insbesondere der Mean Squared Error (MSE) oder die binäre Cross Entropy (BCE) hierfür geeignet sind. Somit gehen die beiden Größen

$$l_D(T) = MSE\left(C(T), B_T\right) \; bzw. l_D(T) = BCE\left(C(T), B_T\right)$$

und

$$l_D(A) = MSE\left(C(A), B_A\right) \; bzw. l_D(A) = BCE\left(C(A), B_A\right)$$

in die Verlustfunktion des Diskriminatornetzwerks **3** ein, so dass beispielsweise DF = $l_D(T) + l_D(A)$ gewählt werden kann.

[0050] Zum Training des Generatornetzwerks **2** wird eine Generatorfehlerfunktion verwendet, um einen Generatorfehlerwert, der aus zwei Teilen besteht, wobei ein erster Teil einem Abweichungsmaß $\mathbf{l_G}$ zwischen der Güte **C** des Ausgangsbilds $\mathbf{A_T}$ entspricht, das basierend auf einem zum Training angelegten Eingangsbild $\mathbf{E_T}$ generiert wurde, und einem Bewertungslabel **B**, das ein vollständiges Erreichen des zweiten Darstellungsstils angibt, insbesondere einem Bewertungslabel $\mathbf{B_T}$, das Trainingsbildern T für das Training des Diskriminatornetzwerks **3** vorgegeben wird, vorzugsweise einem Bewertungslabel von 1.

[0051] Zur Bestimmung des Abweichungsmaßes $\mathbf{l_G}$ mithilfe der Generatorfehlerfunktion kann man jede geeignete Funktion zur Abstandsbewertung annehmen, wobei insbesondere der Mean Squared Error (MSE) oder die binäre Cross Entropy (BCE) hierfür geeignet sind.

$$l_G = MSE\left(D\left(A_T\right), B_T\right) \; bzw. l_G = BCE\left(D\left(A_T\right), B_T\right)$$

wobei $A_T = G(E_T)$ entspricht.

[0052] Der zweite Teil der Generatorfehlerfunktion entspricht einer Ähnlichkeitsgröße S, die in einem Ähnlichkeitsblock **6** mithilfe einer Ähnlichkeitsbewertungsfunktion ermittelt wird. Die Ähnlichkeitsbewertungsfunktion berechnet basierend auf dem für das Training verwendeten Eingangsbild $\mathbf{E_T}$ des ersten Darstellungsstils und dem entsprechend durch das Generatornetzwerk **2** generierte Ausgangsbild $\mathbf{A_T}$ des zweiten Darstellungsstils ein Maß für eine strukturelle Ähnlichkeit der beiden Bilder. Insbesondere kann als eine Ähnlichkeitsbewertungsfunktion eine Funktion vorgesehen sein, die bei einer hohen strukturellen Ähnlichkeit einem Wert nahe 1 und bei keinerlei struktureller Ähnlichkeit nahe -1 annimmt. Geeignet als Ähnlichkeitsbewertungsfunktion ist beispielsweise, eine sogenannte SSIM-Funktion zu wählen, die einen Index struktureller Ähnlichkeit angibt oder eine darauf aufbauende MSSIM, wie beispielsweise aus Zhou Wang et al., „Image Quality Assessment: From Error Visibility to Structural Similarity“, IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 13, NO. 4, APRIL 2004, Seite 600 - 612 bekannt, verwendet.

[0053] In dem Flussdiagramm der **Fig. 3** wird anschaulich das Trainingsverfahren für das erste neuronale Netz des Generatornetzwerks **2** beschrieben, so dass das trainierte Generatornetzwerk **2** zur Änderung eines Darstellungsstils eines Eingangsbildes E verwendet werden kann. Dabei wird initial von einer Anfangsparametrisierung des ersten neuronalen Netzes mit den Generatorparametern $\mathbf{\theta_G}$ und dem zweiten neuronalen Netz mit den Diskriminatorparametern $\mathbf{\theta_D}$ ausgegangen.

[0054] In Schritt **S1** wird basierend auf einem zum Training bereitgestellten Eingangsbild $\mathbf{E_T}$ eines ersten Darstellungsstils ein Ausgangsbild $\mathbf{A_T}$ eines zweiten Darstellungsstils berechnet: $A_T = G_{\theta\,D}(E_T)$.

[0055] Mithilfe des Diskriminatornetzwerks **3** wird in Schritt **S2** eine Güte

$$C\left(A_T\right) = D_{\theta_D}\left(A_T\right)$$

des von dem Generatornetzwerk **2** generierten Ausgangsbildes $\mathbf{A_T}$ und daraus das Abweichungsmaß

$$I_G = MSE\left(C\left(A_T\right), B_T\right)\ bzw. I_G = BCE\left(C\left(A_T\right), B_T\right)$$

ermittelt.

[0056] In Schritt **S3** wird die Ähnlichkeitsgröße S zwischen dem zum Training bereitgestellten Eingangsbild $\mathbf{E_T}$ und dem entsprechenden Ausgangsbild $\mathbf{A_T}$ berechnet:

$$S = SSIM\left(A_T, E_T\right)$$

[0057] Basierend auf dem Abweichungsmaß $\mathbf{I_G}$ und dem Ähnlichkeitsmaß S wird mithilfe der Generatorfehlerfunktion GFF in Schritt **S4** ein Generatorfehlerwert GF für das generierte Ausgangsbild $\mathbf{A_T}$ ermittelt.

$$GF = \mathrm{GFF}\left(I_G, \mathrm{S}\right)$$

[0058] Beispielsweise kann der Generatorfehlerwert GF bestimmt sein als: $GF = I_G + k*S$, wobei der Optimierungsfaktor k entsprechend empirisch gewählt werden kann und insbesondere zwischen -1...-3 gewählt werden kann, wenn $\mathbf{I_G} \in \{0; 1\}$ und $S \in \{-1; 1\}$ sind.

[0059] Basierend auf dem Generatorfehlerwert GF wird in Schritt **S5** ein Lernschritt für das erste neuronale Netz des Generatornetzwerk **2** durchgeführt, insbesondere basierend auf einem Backpropagation-Verfahren. Dadurch werden die Generatorparameter $\mathbf{\theta_G}$ basierend auf den partiellen Ableitungen $dGF/d\theta_G$ aktualisiert.

[0060] Gegebenenfalls können die Schritte **S1** bis **S5** des Trainings des Generatornetzwerks **2** mit demselben oder einem anderen zum Training bereitgestellten Eingangsbild $\mathbf{E_T}$ wiederholt werden.

[0061] Nun beginnt das Training des Diskriminatornetzwerks **3**. In einem nachfolgenden Schritt **S6** wird für ein oder mehrere vorgegebene Trainingsbilder $T_{1..n}$ in dem zweiten Darstellungsstil eine Güte $C(T_{1..n})$ entsprechend dem aktuellen Trainingszustand des Diskriminatornetzwerks **3** ermittelt und daraus in Schritt **S7** das bzw. die Abweichungsmaße $I_D(T_{1..n})$ bestimmt:

$$I_D\left(T\right) = MSE\left(C\left(T\right), B_T\right)\ bzw. I_D\left(T\right) = BCE\left(C\left(T\right), B_T\right)$$

[0062] Weiterhin wird in Schritt **S8** für ein oder mehrere zuletzt generierte Ausgangsbilder $A_{1..m}$ entsprechend eine Güte $C(A_{1..m})$ ermittelt und daraus in Schritt **S9** das bzw. die Abweichungsmaße $I_D(A_{1..m})$ bestimmt:

$$I_D\left(A\right) = MSE\left(D_\theta\left(A\right), B_A\right)\ bzw. I_D\left(A\right) = BCE\left(D_\theta\left(A\right), B_A\right)$$

[0063] In einem nächsten Schritt **S10** wird ein Diskriminatorfehlerwert DF beispielsweise entsprechend folgender Formel ermittelt:

$$DF = \left(\sum_n I_D\left(A\right) + \sum_m I_D\left(T\right)\right) I\left(n+m\right)$$

[0064] Basierend auf dem Diskriminatorfehlerwert DF kann in Schritt **S11** ein Lernschritt für das zweite neuronale Netz des Diskriminatornetzwerks **3** durchgeführt werden. Dadurch werden die Diskriminatorparameter θ_D in einem Backpropagation-Verfahren entsprechend mithilfe der partiellen Ableitungen $dDF/d\theta_D$ aktualisiert.

[0065] Selbstverständlich kann das Backpropagation-Verfahren auch nur basierend auf einem Trainingsbild T und/oder einem Ausgangsbild A durchgeführt werden.

[0066] Zusätzlich können für das Trainieren des Diskriminatornetzwerks **3** nicht nur generierte Bilder in dem zweiten Darstellungsstil, sondern auch sonstige Trainingsbilder in dem ersten Darstellungsstil mit der Güte **0** (oder nahe 0) verwendet werden. Hierdurch erleichtert man es dem Diskriminator ggf. die Unterschiede zwischen beiden Domänen besser zu lernen.

[0067] Nun wird in Schritt **S12** eine Abbruchbedingung überprüft. Ist die Abbruchbedingung nicht erfüllt (Alternative: Nein), wird das Verfahren mit Schritt **S1** fortgesetzt, anderenfalls (Alternative: Ja) wird das Verfahren mit Schritt **S13** fortgesetzt. Eine Abbruchbedingung kann beispielsweise das Erreichen einer Anzahl von Durchgängen sein oder das Erreichen eines vorbestimmten Diskriminatorfehlerwerts **DF** und/oder Generatorfehlerwerts GF, oder das Erreichen einer vorbestimmten Güte C(A) der vom Generatornetzwerk **2** generierten Ausgangsbilder A.

[0068] Der Schritt **S13** stellt nun das Generatornetzwerk **2** als System zur Umsetzung eines Eingangsbildes E eines ersten Darstellungsstils bzw. einer ersten Domäne in ein Ausgangsbild A eines zweiten Darstellungsstils bzw. einer zweiten Domäne dar.

[0069] Das oben beschriebene Verfahren kann in vielfältiger Weise modifiziert werden. So ist es möglich, dass die Diskriminatorparameter θ_D und Generatorparameter θ_G nur unter bestimmten Bedingungen aktualisiert werden, z. B. abhängig von dem aktuellen Diskriminatorfehlerwert **DF** zum Training des Diskriminatornetzwerks **3** bzw. des Generatorfehlerwerts **GF** zum Training des Generatornetzwerks **2**. Auch die Größe der Batches für das Training des Diskriminatornetzwerks **3** oder des Generatornetzwerks **2** kann variiert werden.

[0070] Weiterhin kann bei der Diskriminatorfehlerfunktion **DFK** des Diskriminatornetzwerks **3** noch ein Eingangsbild-Abweichungsmaß additiv hinzugefügt werden, die eine Abweichung der Güte **C** des Eingangsbildes von einem Bewertungslabel B_A für ein unechtes Bild, d. h. eines vom Generatornetzwerk generierten Ausgangsbildes **A** angibt. Dadurch kann die Stabilität des Trainings erhöht werden.

[0071] Das trainierte Generatornetzwerk **2** kann dann dazu benutzt werden, aus Eingangsbildern **E**, die über eine Skript-basierte Beschreibung erstellt wurden, die z. B. Verkehrssituationen zeigen, Eingangsbilder **E** eines ersten Darstellungsstils zu erzeugen. Ist das Generatornetzwerk **2** basierend auf Bildern des ersten Darstellungsstils und photorealistischen Bildern von Verkehrssituationen trainiert worden, so lassen sich den künstlich erzeugten Eingangsbildern **E** photorealistische Bilder zuordnen, die eine entsprechende Verkehrssituation darstellen. Dadurch können mithilfe des Generatornetzwerks **2** beliebig viele photorealistische Bilder erstellt werden, die gewünschte Verkehrssituationen abbilden.

[0072] Das Generatornetzwerk **2** lässt sich auch in umgekehrter Weise trainieren, um photorealistische Bilder in synthetische Bilder umzuwandeln, um beispielsweise Reflexionen oder Ähnliches aus den photorealistischen Bildern zu entfernen, wenn beispielsweise ein Klassifikator synthetische Bilder besser klassifizieren kann als photorealistische Bilder.

[0073] Weiterhin kann das obige System auch so trainiert werden, um aus photorealistischen Bildern segmentierte Bilder zu erstellen, wobei in diesem Fall die photorealistischen Bilder dem ersten Darstellungsstil entsprechen und die segmentierten Bilder den Bildern des zweiten Darstellungsstils.

ZITATE ENTHALTEN IN DER BESCHREIBUNG

Diese Liste der vom Anmelder aufgeführten Dokumente wurde automatisiert erzeugt und ist ausschließlich zur besseren Information des Lesers aufgenommen. Die Liste ist nicht Bestandteil der deutschen Patent- bzw. Gebrauchsmusteranmeldung. Das DPMA übernimmt keinerlei Haftung für etwaige Fehler oder Auslassungen.

Zitierte Nicht-Patentliteratur

- Zhou Wang et al., „Image Quality Assessment: From Error Visibility to Structural Similarity“, IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 13, NO. 4, APRIL 2004, Seite 600 - 612 [0052]

Patentansprüche

1. Verfahren zum Trainieren eines ersten neuronalen Netzes zur Umwandlung eines Eingangsbildes (E) einer ersten Domäne in ein Ausgangsbild (A) einer zweiten Domäne, wobei das Training auf für das Training bereitgestellten Eingangsbildern (E) der ersten Domäne und Trainingsbildern (T) der zweiten Domäne durchgeführt wird; mit folgenden Schritten:
- Bereitstellen eines GAN-Netzwerks mit einem Generatornetzwerk (2), das das erste neuronale Netz umfasst, und einem Diskriminatornetzwerk (3), das ein zweites neuronales Netz umfasst;
- Trainieren des Diskriminatornetzwerks (3) basierend auf einem Diskriminatorfehlerwert (DF) und einem oder mehreren Trainingsbildern (T) und/oder einem oder mehreren Ausgangsbildern (A), die durch Verarbeiten eines oder mehrerer der Eingangsbilder durch das Generatornetzwerk (2) erzeugt werden, wobei der Diskriminatorfehlerwert (DF) abhängig von einer jeweiligen Güte (C) des einen oder der mehreren Trainingsbilder (T) und/oder des einen oder der mehreren Ausgangsbilder bestimmt wird;
- Trainieren des ersten neuronalen Netzes des Generatornetzwerks (2) basierend auf einem für das Training bereitgestellten Eingangsbild (E) und einem Generatorfehlerwert (GF), der von einer Güte (C) des von dem Generatornetzwerk (2) abhängig von dem Eingangsbild (E) bereitgestellten Ausgangsbilds (A) und einer Ähnlichkeitsgröße (S) zwischen dem Eingangsbild (E) und dem Ausgangsbild (A) abhängt, die ein Maß für eine strukturelle Ähnlichkeit angibt.

2. Verfahren nach Anspruch 1, wobei das Trainieren des Diskriminatornetzwerks (3) und des Generatornetzwerks (2) gleichzeitig oder wechselweise wiederholt durchgeführt wird, insbesondere mithilfe eines Backpropagation-Verfahrens, bis eine Abbruchbedingung erfüllt ist.

3. Verfahren nach Anspruch 2, wobei die Abbruchbedingung erfüllt ist, wenn eine Anzahl von Durchgängen oder eine vorbestimmte Güte (C) der vom Generatornetzwerk (2) generierten Ausgangsbilder erreicht ist.

4. Verfahren nach einem der Ansprüche 1 bis 3, wobei die Güte (C) des einen oder der mehreren Trainingsbilder (T) und/oder des einen oder der mehreren Ausgangsbilder (A) jeweils durch das Diskriminatornetzwerk (3) bestimmt wird und einer Bewertung entspricht, in welchem Maß es sich bei dem betreffenden Bild um ein Bild der zweiten Domäne handelt.

5. Verfahren nach Anspruch 4, wobei der Diskriminatorfehlerwert (DF) abhängig von einem jeweiligen Abweichungsmaß für die Abweichung zwischen der jeweiligen Güte (C) des betreffenden einen oder der betreffenden mehreren Trainingsbilder (T) und einem Bewertungslabel, das eine jeweilige Zugehörigkeit des einen oder der mehreren Trainingsbilder (T) zu der zweiten Domäne angibt, und/oder abhängig von einem jeweiligen Abweichungsmaß für die Abweichung zwischen der jeweiligen Güte (C) des einen oder der mehreren Ausgangsbilder (A) und einem Bewertungslabel (B), das ein von dem Generatornetzwerk (2) generiertes Ausgangsbild als ein nicht der zweiten Domäne zugehöriges Bild angibt, bestimmt wird, wobei das Abweichungsmaß insbesondere einem Mean Squared Error oder einer binären Cross Entropy entspricht.

6. Verfahren nach einem der Ansprüche 1 bis 5, wobei die Ähnlichkeitsgröße von einem SSIM-Index für eine strukturelle Ähnlichkeit zwischen einem der Eingangsbilder und einem durch das Generatornetzwerk aus dem betreffenden Eingangsbild generierten Ausgangsbild abhängt oder diesem entspricht.

7. Verfahren nach einem der Ansprüche 1 bis 6, wobei das erste und/oder das zweite neuronale Netz als faltende neuronale Netze (convolutional neural networks) ausgebildet sind, wobei insbesondere das erste und/oder das zweite neuronale Netz eine Hintereinanderschaltung von einigen Convolutional Layer-Blöcken, einigen ResNet-Blöcken und einigen Deconvolutional-Blöcken aufweisen, wobei insbesondere jeder der Blöcke als eine Aktivierungsfunktionen eine ReLU-, leaky-ReLU-, tanh- oder Sigmoid-Funktion enthält.

8. Verfahren nach einem der Ansprüche 1 bis 7, wobei der Generatorfehlerwert (GF) von einem jeweiligen Abweichungsmaß für die Abweichung zwischen der jeweiligen Güte (C) des von dem Generatornetzwerk (2) abhängig von dem Eingangsbild (E) bereitgestellten Ausgangsbilds (A) und einem Bewertungslabel (B), das ein Bild der zweiten Domäne angibt, abhängt, wobei das Abweichungsmaß insbesondere einem Mean Squared Error oder einer binären Cross Entropy entspricht.

9. Verfahren nach einem der Ansprüche 1 bis 8, wobei das Training des Diskriminatornetzwerks und/oder des Generatornetzwerks nur dann durchgeführt wird, wenn eine von dem aktuellen Diskriminatorfehlerwert (DF) und/oder von dem Generatorfehlerwert (GF) abhängige Bedingung erfüllt ist.

10. Verfahren zum Bereitstellen einer Steuerung für ein technisches System, insbesondere für einen Roboter, ein Fahrzeug, ein Werkzeug oder eine Werkmaschine, wobei das Verfahren zum Trainieren eines ersten neuronalen Netzes nach einem der Ansprüche 1 bis 9 ausgeführt wird, wobei das trainierte erste neuronale Netz verwendet wird, um Trainingsbilder zu erzeugen, mit denen die Steuerung, die insbesondere ein neuronales Netz enthält, trainiert wird.

11. Verfahren nach Anspruch 10, wobei das technische System mithilfe der Steuerung betrieben wird.

12. Verwenden eines ersten neuronalen Netzes, das entsprechend einem Verfahren nach einem der Ansprüche 1 bis 9 trainiert ist, zum Generieren von photorealistischen Ausgangsbildern in einer zweiten Domäne abhängig von vorgegebenen Eingangsbildern (E) in einer ersten Domäne, die insbesondere über eine Skript-basierte Beschreibung erstellt werden

13. Verwendung nach Anspruch 12, wobei die erzeugten photorealistischen Ausgangsbilder (A) als künstliche Kamerabilder zum Herstellen eines Klassifikators für Umgebungssituationen verwendet werden.

14. GAN-Netzwerk zum Trainieren eines ersten neuronalen Netzes zur Umwandlung eines Eingangsbildes (E) einer ersten Domäne in ein Ausgangsbild (A) einer zweiten Domäne, wobei das Training auf für das Training bereitgestellten Eingangsbildern (E) der ersten Domäne und Trainingsbildern (T) der zweiten Domäne durchgeführt wird, wobei das GAN-Netzwerk ein Generatornetzwerk (2), das das erste neuronale Netz umfasst, und ein Diskriminatornetzwerk (3) aufweist, das ein zweites neuronales Netz umfasst, wobei das GAN-Netzwerk ausgebildet ist, um
- das Diskriminatornetzwerk (3) basierend auf einem Diskriminatorfehlerwert (DF) und einem oder mehreren Trainingsbildern (T) und/oder einem oder mehreren Ausgangsbildern (A), die durch Verarbeiten eines oder mehrerer der Eingangsbilder durch das Generatornetzwerk (2) erzeugt werden, zu trainieren, wobei der Diskriminatorfehlerwert (DF) abhängig von einer jeweiligen Güte (C) des einen oder der mehreren Trainingsbilder (T) und/oder des einen oder der mehreren Ausgangsbilder bestimmt wird; und
- das Generatornetzwerk (2) basierend auf einem für das Training bereitgestellten Eingangsbild (E) und einem Generatorfehlerwert (GF) zu trainieren, der von einer Güte (C) des von dem Generatornetzwerk (2) abhängig von dem Eingangsbild (E) bereitgestellten Ausgangsbilds (A) und einer Ähnlichkeitsgröße (S) zwischen dem Eingangsbild (E) und dem Ausgangsbild (A) abhängt, die ein Maß für eine strukturelle Ähnlichkeit angibt.

15. Computerprogramm mit Programmcodemitteln, das dazu eingerichtet ist, ein Verfahren nach einem der Ansprüche 1 bis 9 auszuführen, wenn das Computerprogramm auf einer Recheneinheit, insbesondere einer mobilen Recheneinheit, ausgeführt wird.

16. Maschinenlesbares Speichermedium mit einem darauf gespeicherten Computerprogramm nach Anspruch 15.

Es folgen 3 Seiten Zeichnungen

Anhängende Zeichnungen

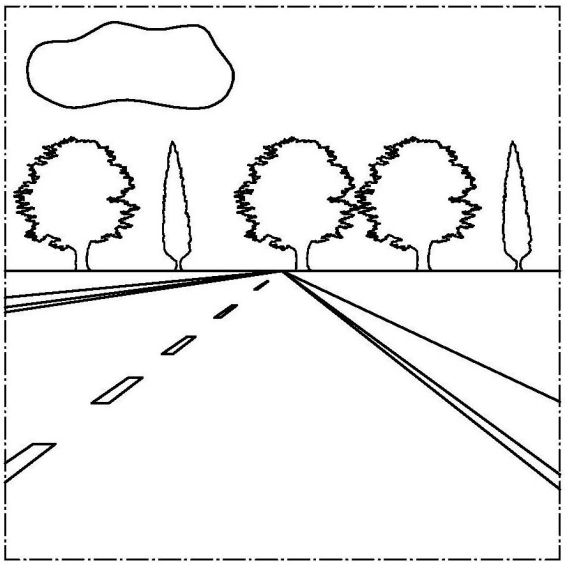




Fig. 1a

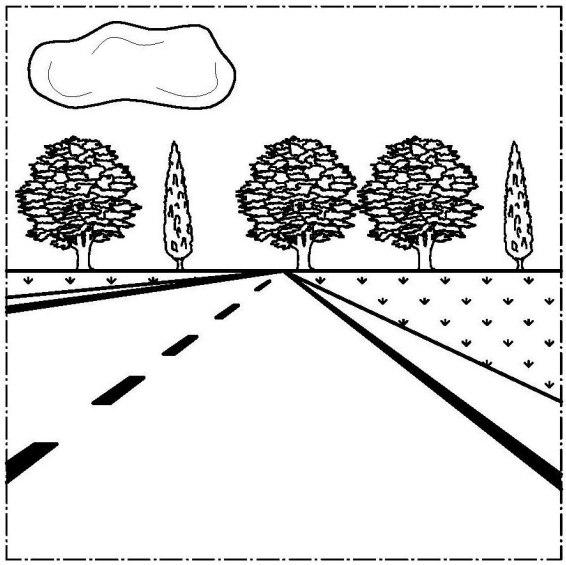




Fig. 1b

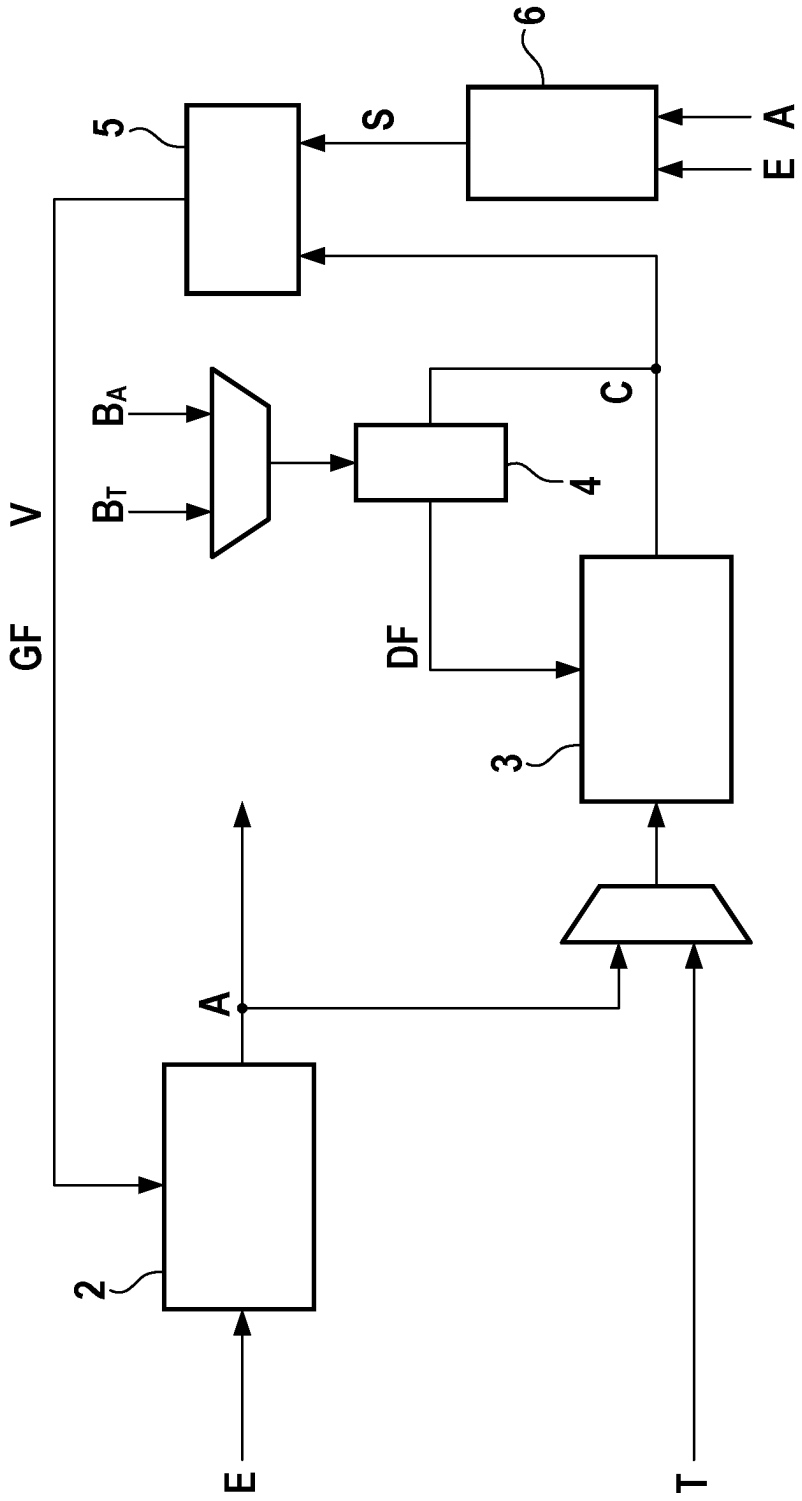


Fig. 2

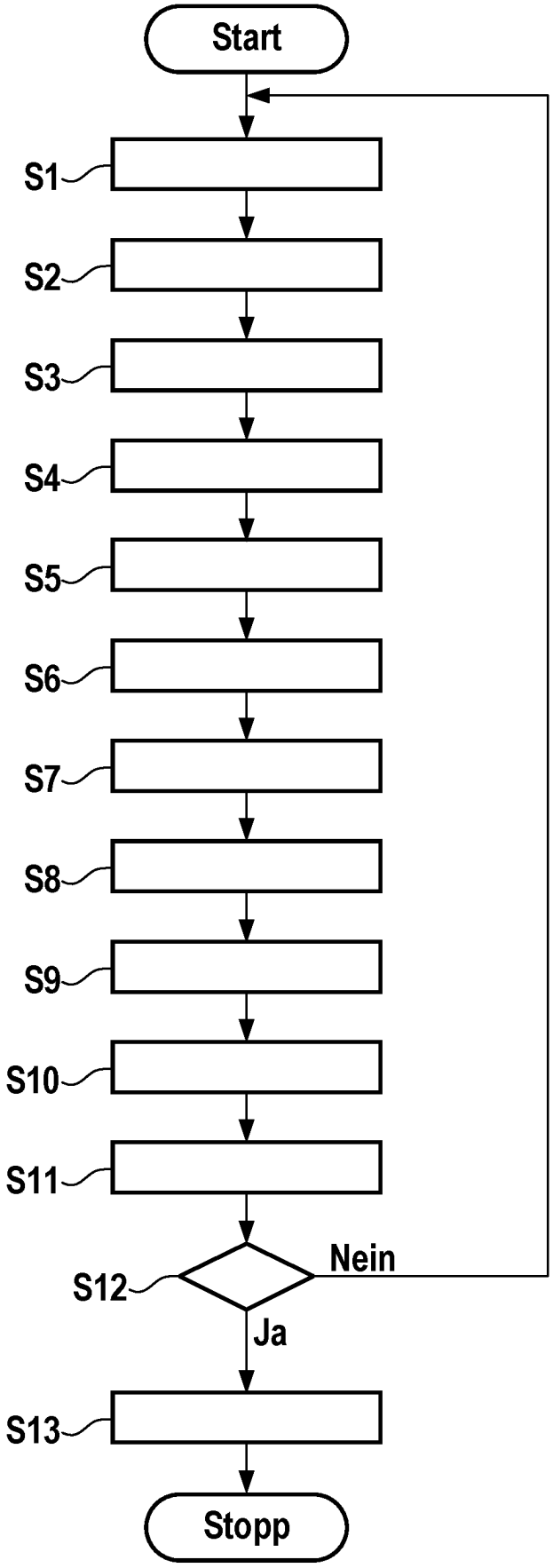


Fig. 3