

Veröffentlichung

der internationalen Anmeldung mit der
(87) Veröffentlichungs-Nr.: **WO 2019/027598**
in der deutschen Übersetzung (Art. III § 8 Abs. 2
IntPatÜG)

(51) Int Cl.: **B64C 39/02** (2006.01)
B64D 47/00 (2006.01)
G06F 8/65 (2018.01)

(21) Deutsches Aktenzeichen: **11 2018 003 945.5**
 (86) PCT-Aktenzeichen: **PCT/US2018/040070**
 (86) PCT-Anmeldetag: **28.06.2018**
 (87) PCT-Veröffentlichungstag: **07.02.2019**
 (43) Veröffentlichungstag der PCT Anmeldung
 in deutscher Übersetzung: **07.05.2020**

(30) Unionspriorität:		
62/497,300	02.08.2017	US
15/676,283	14.08.2017	US

(74) Vertreter:
**Maucher Jenkins Patentanwälte & Rechtsanwälte,
80538 München, DE**

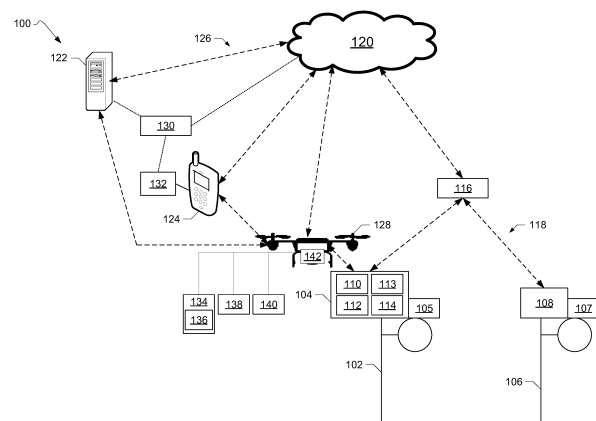
(71) Anmelder:
Intel Corporation, Santa Clara, Calif., US

(72) Erfinder:
Sharma, Dhaval Vinayak, Bangalore, IN

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen.

(54) Bezeichnung: **VERFAHREN UND VORRICHTUNGEN ZUR BEREITSTELLUNG VON DRAHTLOS-AKTUALISIERUNGEN ZU SENSORKNOTEN DES INTERNETS DER DINGE**

(57) Zusammenfassung: Hierin sind Beispielverfahren und Vorrichtungen zum Bereitstellen von Drahtlos-Aktualisierungen für IoT-Sensorknoten offenbart. Ein unbemanntes Beispiel-Fluggerät weist einen Aktualisierungsübermittler auf, um auf eine Firmwareaktualisierung, welche einem Sensorknoten in einem Netzwerk zu übermitteln ist, zuzugreifen. Der Sensorknoten ist mit einem Objekt verbunden. Das unbemannte Beispiel-Fluggerät weist eine Kamera auf, um Bilddaten zu erzeugen, sowie einen Identifikator, um das Objekt basierend auf den Bilddaten zu identifizieren. Der Aktualisierungsübermittler übermittelt die Firmwareaktualisierung basierend auf der Identifikation des Objekts zum Sensorknoten.



Beschreibung**GEBIET DER OFFENBARUNG**

[0001] Diese Offenbarung betrifft im Allgemeinen Sensorknoten in einem Internetder-Dinge-Netzwerk und insbesondere Verfahren und Vorrichtungen zur Bereitstellung von Drahtlos-Aktualisierungen zu Sensorknoten des Internets der Dinge.

ALLGEMEINER STAND DER TECHNIK

[0002] Das Internet der Dinge (IoT) umfasst ein Netzwerk von kommunikativ verbundenen Objekten (z.B. Vorrichtungen), welches Sensorknoten aufweist, die Daten sammeln und die Daten drahtlos über das Netzwerk zur Analyse, zum Beispiel durch eine cloudbasierte Vorrichtung, übertragen. Die Sensorknoten können in einem Energiesparmodus arbeiten, um den Energieverbrauch, zum Beispiel während der Datenübertragung über große Entfernungen, zu minimieren.

[0003] Die Sensorknoten weisen Firmware auf, welche regelmäßige Aktualisierungen erfordern kann. Drahtlos-Aktualisierungen ermöglichen es, Firmware-Aktualisierungen an Vorrichtungen in einem IoT-Netzwerk, welche mit einer Kabelverbindung schwierig zu erreichen sein könnten, zu übermitteln. Das drahtlose Übermitteln einer FirmwareAktualisierung zu einem Sensorknoten in einem Energiesparmodus in einem IoT-Netzwerk kann aufgrund langsamer Geschwindigkeiten der Drahtlosverbindung jedoch ineffizient sein.

Figurenliste

Fig. 1 zeigt ein Beispielsystem aufweisend ein IoT-Netzwerk aufweisend einen oder mehrere Sensorknoten und ein unbemanntes Fluggerät (UAV) zum Übermitteln von Aktualisierungen zu den Sensorknoten im Einklang mit den hierin offenbarten Lehren.

Fig. 2 ist ein Blockdiagramm einer Beispielumsetzung des UAV-Managers von **Fig. 1**.

Fig. 3 ist ein Blockdiagramm einer Beispielumsetzung des Sensorknotenaktualisierungsmanagers von **Fig. 1**.

Fig. 4 ist ein Ablaufdiagramm, welches maschinenlesbare Beispielanweisungen darstellt, die ausgeführt werden können, um den Beispiel-Sensorknotenaktualisierungsmanager von **Fig. 1** und/oder 3 umzusetzen.

Fig. 5 ist ein Ablaufdiagramm, welches zweite maschinenlesbare Beispielanweisungen darstellt, die ausgeführt werden können, um den Beispiel-Sensorknotenaktualisierungsmanager von **Fig. 1** und/oder 3 umzusetzen.

Fig. 6 ist ein Ablaufdiagramm, welches maschinenlesbare Beispielanweisungen darstellt, die ausgeführt werden können, um den Beispiel-UAV-Manager von **Fig. 1** und/oder 2 umzusetzen.

Fig. 7 zeigt eine erste Beispiel-Prozessorplattform, die dafür gestaltet ist, eine oder mehrere der Beispielanweisungen von **Fig. 4** und/oder 5 auszuführen, um den Beispiel-Sensorknotenaktualisierungsmanager von **Fig. 1** und/oder 3 umzusetzen.

Fig. 8 zeigt eine zweite Beispiel-Prozessorplattform, die dafür gestaltet ist, eine oder mehrere der Beispielanweisungen von **Fig. 6** auszuführen, um den Beispiel-UAV-Manager von **Fig. 1** und/oder 2 umzusetzen.

[0004] Die Figuren sind nicht maßstabsgetreu. Soweit wie möglich werden in sämtlichen Zeichnungen und der zugehörigen schriftlichen Beschreibung dieselben Bezugsziffern zur Bezeichnung derselben oder ähnlicher Teile verwendet.

AUSFÜHRLICHE BESCHREIBUNG

[0005] Das Internet der Dinge (IoT) umfasst ein Netzwerk von Objekten mit eingebetteten Sensoren, welche es ermöglichen, die Objekte mit dem Internet zu verbinden und durch die Sensoren gesammelte Daten zur Analyse auf das Internet zu übertragen, um Anweisungen zu empfangen, etc. Beispiel Objekte eines IoT-Netzwerks können Benutzervorrichtungen, wie zum Beispiel Smartphones, in Menschen implantierte medizinische Vorrichtungen und/oder industrielle Ausrüstung, wie zum Beispiel intelligente Motoren, sein. Die Objekte weisen damit verbundene (zum Beispiel eingebettete) Sensorknoten auf, welche Sensoren zum Sammeln von Daten aufweisen. Zum Beispiel kann ein Sensor einer Straßenlampe Daten über die Helligkeit und/oder die Dauer, für welche die Lampe eingeschaltet ist, sammeln, und die Daten über das Netzwerk zur Analyse zu einer cloudbasierten Vorrichtung senden. Als weiteres Beispiel kann die Straßenlampe Sensoren umfassen, welche Temperatur- und Feuchtigkeitsdaten der Umgebung, in welcher sich die Straßenlampe befindet, sammeln. In manchen Beispielen können die Sensorknoten in einem IoT-Netzwerk mit einem Master-Knoten verbunden sein, welcher von den Sensorknoten empfangene Daten zum Beispiel zu einer cloudbasierten Vorrichtung übermitteln kann.

[0006] Ein Sensorknoten kann Hardware aufweisen, wie zum Beispiel eine Stromquelle, eine Steuerung (z.B. einen Mikrokontroller), einen Sensor und einen drahtlosen Senderempfänger. Der Senderempfänger übermittelt durch die Sensoren gesammelte Daten zur cloudbasierten Vorrichtung in einem IoT-Netzwerk (z.B. über den Master-Knoten) und kann über

die Cloud und/oder den Master-Knoten, zum Beispiel von einer weiteren Vorrichtung im IoT-Netzwerk (z.B. einer Benutzervorrichtung), Anweisungen empfangen. Beispiel-Sensorknoten in einem IoT-Netzwerk umfassen energiesparende Sensorknoten großer Reichweite. Solche Beispiel-Sensorknoten sind in der Lage, Daten über große Entfernungen (zum Beispiel über ein Maschen-Netzwerk, ein drahtloses Netzwerk geringer Geschwindigkeit) zu übertragen, während sie den Energieverbrauch, zum Beispiel um die Batterielebensdauer zu verbessern, minimieren. Jedoch erzielen solche Sensorknoten oftmals eine Übertragung über große Entfernungen, indem sie die Datenrate, mit welcher die Daten übertragen werden, verringern (z.B. auf eine Drahtlosgeschwindigkeit von weniger als 1 kbps). Zum Beispiel können die Knoten Daten über Sub-1-GHz-Frequenzbänder übertragen. Als weiteres Beispiel können die Knoten Daten über LoRaWAN™ übertragen, ein energiesparendes Weitverkehrsnetzwerk, welches für eine Datenübertragung von einem Knoten zu einem Server optimiert ist, jedoch eine verringerte Bandbreitenverfügbarkeit für eine Datenübertragung vom Server zum Knoten (zum Beispiel eine acht Mal geringere Bandbreitenverfügbarkeit) aufweist.

[0007] In manchen Beispielen können die Sensorknoten Firmware aufweisen, welche Aktualisierungen benötigt, um die Leistung aufrechtzuerhalten, zusätzliche Merkmale/Funktionen zu installieren und/oder Fehler zu beheben. In einem IoT-Netzwerk können die Sensorknoten in einer Umgebung angeordnet sein, in welcher das Einrichten einer verdrahteten Verbindung zum Aktualisieren der Firmware nicht machbar ist. Ferner kann es eine Vielzahl von Sensorknoten geben (z.B. Hunderte), welche aktualisiert werden müssen. Drahtlos-Firmware- (FOTA-) Aktualisierungen können dazu verwendet werden, Firmwareaktualisierungen zu übertragen, ohne dass eine verdrahtete Verbindung erforderlich ist. In einigen bekannten Beispielen kann der Master-Knoten eine FOTA-Aktualisierung im gesamten Netzwerk von Sensorknoten über ein drahtloses Netzwerk geringer Geschwindigkeit oder ein Maschen-Netzwerk, welches die Sensorknoten und den Master-Knoten verbindet, übertragen. Allerdings ist das Aktualisieren der Firmware mit geringen Datengeschwindigkeiten über mehrere Sensorknoten zeitaufwändig und ineffizient. Ferner können große Datenmengen, welche während der FOTA-Aktualisierung übertragen werden, die Knotenstabilität im Netzwerk beeinträchtigen.

[0008] Manche Mikrokontroller weisen drahtlose Dualband-Funkgeräte auf, welche zwischen verschiedenen Drahtlosgeschwindigkeitsfunktionen wechseln können. Bei ihrer Verwendung in einem Sensorknoten eines IoT-Netzwerks stellen solche Beispiel-Mikrokontroller Kommunikation mit geringen Geschwindigkeiten zum Beispiel über drahtlose Kommunika-

tion im Sub-1-GHz-Frequenzband bereit, während zum Beispiel ein Sensor des Sensorknotens Sensordaten sammelt und/oder die Sensordaten über große Entfernungen zu anderen Sensorknoten, dem Master-Knoten, etc. (z.B. über ein Maschen-Netzwerk) übertragen werden. Solche Beispiel-Mikrokontroller unterstützen auch Kommunikation mit höheren Geschwindigkeiten zum Beispiel durch drahtlose Kommunikation über das 2,4-GHz-Frequenzband oder durch Niedrigenergie-Bluetooth (BLE). Obwohl Daten mit höheren Geschwindigkeiten unter Verwendung eines Dualband-Mikrokontrollers über das 2,4-GHz-WiFi-Frequenzband oder BLE ausgetauscht werden können, gibt es jedoch Einschränkungen in Bezug auf die Entfernungen, über welche Hochgeschwindigkeitsdatenübertragungen erfolgen können. Im Zusammenhang mit einem IoT-Netzwerk muss ein Sensorknoten, welcher einen Dualband-Mikrokontroller enthält, zum Beispiel in der Nähe (z.B. 10 - 30 Meter) eines Master-Knotens angeordnet sein, um über das 2,4-GHz-WiFi-Frequenzband oder über BLE eine Firmwareaktualisierung vom Master-Knoten zu empfangen. Solche Einschränkungen hinsichtlich der Nähe können in einem IoT-Netzwerk, welches viele Sensorknoten angeordnet in einem großen Gebiet (z.B. einer Stadt oder einem Wald) aufweist, nicht zweckmäßig sein. Obwohl die Sensorknoten über das 2,4-GHz-WiFi-Frequenzband oder BLE manuell aktualisiert werden können, indem Verbindungen mit jedem der Sensorknoten in der erforderlichen Nähe (z.B. 10 m) zur Einleitung der Aktualisierungen eingerichtet werden, wären solche Aktualisierungen zeitaufwändig und in manchen Beispielen aufgrund der Anzahl von Sensorknoten, welche im IoT-Netzwerk aktualisiert werden müssen, nicht praktikabel.

[0009] Unbemannte Fluggeräte (UAV), wie zum Beispiel Drohnen (z.B. rotierende Drohnen), können eine bestimmte Zeit lang über einem bestimmten Gebiet schweben. Einige UAV weisen drahtlose Verbindungsfunktionen, wie zum Beispiel WiFi (z.B. über das 2,4-GHz-Frequenzband) und/oder Bluetooth-Verbindungen, auf. Somit können manche UAV drahtlose Kommunikation mit anderen Vorrichtungen aufbauen, um eine Hochgeschwindigkeitsdatenübertragung (z.B. im Vergleich zur Kommunikation über das Sub-1-GHz-Frequenzband) bereitzustellen. Manche Beispiel-UAV weisen Kameras auf, welche Bilddaten erzeugen, sowie Steuerungen (z.B. Prozessoren) zum Speichern und/oder Analysieren der Bilddaten und/oder sonstiger Daten.

[0010] Hierin offenbarte Beispielsysteme und Verfahren übermitteln (einem) Sensorknoten in einem IoT-Netzwerk Firmwareaktualisierung(en) durch (ein) UAV, welche/s Drahtlosverbindungen mit dem/den Sensorknoten einrichtet. Einige Beispiele weisen einen Sensorknoten auf, welcher einen drahtlosen Dualband-Mikrokontroller, welcher Kommunikation über große Entfernungen bei niedrigeren Frequen-

zen (z.B. über das Sub-1-GHz-Frequenzband durch ein Maschen-Netzwerk oder ein Drahtlos-Netzwerk mit geringer Geschwindigkeit) sowie Kommunikation über kurze Entfernungen mit höherer Geschwindigkeit bei höheren Frequenzen (z.B. über das 2,4-GHz-Frequenzband und/oder das 5-GHz-Frequenzband; durch WiFi, BLE), bereitstellt, aufweist. In einigen Beispielen arbeitet der Sensorknoten in einem ersten Drahtlosmodus, wie zum Beispiel einem Energiespar-Drahtlosmodus (z.B. Sub-1-GHz), um Sensordaten zu sammeln und um die Sensordaten zum Beispiel zu einem Master-Knoten im IoT-Netzwerk zu übertragen. Wenn der Sensorknoten aktualisiert werden soll, fliegt ein UAV zu einer Position in der Nähe des Sensorknotens und sendet eine Anweisung an den drahtlosen Dualband-Mikrokontroller des Sensorknotens, auf einen zweiten Drahtlosmodus, wie zum Beispiel einen Hochgeschwindigkeitsdrahtlosmodus (z.B. 2,4-GHz, BLE) umzuschalten. Hierin offenbarte Beispiel-UAV weisen einen 2,4-GHz-Senderempfänger (und/oder einen anderen drahtlosen Hochgeschwindigkeits-Senderempfänger) auf, welcher sich mit dem Sensorknoten verbindet, wenn sich der Sensorknoten im Hochgeschwindigkeitsdrahtlosmodus befindet, um die Firmwareaktualisierung mit schnelleren Datenraten zum Sensorknoten zu übertragen, als beim Aktualisieren des Sensorknotens im Energiespar-Drahtlosmodus erzielt würden. In manchen Beispielen schwebt das UAV mindestens während eines Zeitraums, der benötigt wird, um den Sensorknoten zu aktualisieren, rund um den Sensorknoten.

[0011] Hierin offenbarte Beispiel-UAV weisen Mittel auf, um ein IoT-Objekt im IoT-Netzwerk zu erkennen, welches einen zu aktualisierenden Sensorknoten aufweist. In manchen Beispielen erkennt das UAV den zu aktualisierenden Sensorknoten basierend auf Bilddaten, welche das UAV während des Flugs sammelt, und einer dem Sensorknoten zugewiesenen Kennung. In manchen Beispielen erkennt das UAV den zu aktualisierenden Sensorknoten zusätzlich oder alternativ dazu basierend auf der Stärke von durch den Sensorknoten ausgesendeten Signalen, welche das UAV darüber informieren, dass sich das UAV in der Nähe des zu aktualisierenden Sensorknotens befindet. Somit können in hierin offenbarten Beispielen die Sensorknoten des IoT-Netzwerks effizient durch FOTA-Aktualisierungen, die über ein oder mehrere UAV übertragen werden, aktualisiert werden.

[0012] In manchen hierin offenbarten Beispielen empfängt das UAV von einer einem Betreiber zugeordneten Benutzervorrichtung Anweisungen in Bezug auf den/die zu aktualisierenden Sensorknoten, eine Route, welcher das UAV folgen muss, um den/die Sensorknoten zu erreichen, und die zu übertragende Aktualisierung. In manchen Beispielen werden die Anweisungen durch eine Vorrichtung im IoT-

Netzwerk (z.B. einer Benutzervorrichtung, einen Server, einer cloudbasierten Vorrichtung) automatisch erzeugt. In manchen Beispielen überträgt das UAV Mitteilungen zu einer oder mehreren Vorrichtungen im IoT-Netzwerk und/oder zum Betreiber während der Installation der FOTA-Aktualisierung am Sensorknoten. Die Mitteilung(en) können den Status der Aktualisierung am Sensorknoten, eine Bestätigung, dass die Aktualisierung erfolgreich war, etc. enthalten. In Beispielen, in welchen die FOTA-Aktualisierung erfolgreich war, weist das UAV den drahtlosen Dualband-Mikrokontroller des Sensorknotens an, zur fortgesetzten Sammlung und/oder Übertragung von Sensordaten im gesamten IoT-Netzwerk zurück in den Energiespardrahtlosmodus zu schalten. Somit können in hierin offenbarten Beispielen die Sensorknoten des IoT-Netzwerks effizient durch FOTA-Aktualisierungen, die über ein oder mehrere UAV übertragen werden, aktualisiert werden. In hierin offenbarten Beispielen dienen die UAV als ein wirksames Mittel zum Bewältigen der Anforderungen in Bezug auf Nähe in Zusammenhang mit dem Hochgeschwindigkeitsdrahtlosmodus zum Aktualisieren der Sensorknoten.

[0013] Fig. 1 zeigt ein Beispielsystem **100** aufgebaut im Einklang mit der Lehre dieser Offenbarung zum Übertragen von Drahtlos-Firmware- (FOTA-) Aktualisierungen zu Sensorknoten, welche Objekten in einem IoT-Netzwerk zugeordnet sind. Zu Veranschaulichungszwecken wird das IoT-Netzwerk im Zusammenhang mit Straßenlampen, welche Sensoren zum Sammeln von Daten über die Umwelt, in welcher die Straßenlampen angeordnet sind, wie zum Beispiel Temperatur und Feuchtigkeit, aufweisen, erörtert. Die hierin offenbarten Beispiele können jedoch in irgendeinem IoT-Netzwerk und/oder mit anderen IoT-Objekten (z.B. Benutzervorrichtungen, Gebäuden) verwendet werden. Die Erörterung der Übertragung von Firmwareaktualisierungen an Sensorknoten, welche Straßenlampen zugeordnet sind, als solche dient somit nur veranschaulichenden Zwecken und schränkt diese Offenbarung keinesfalls auf ein bestimmtes Objekt in einem IoT-Netzwerk und/oder ein bestimmtes IoT-Netzwerk ein. Obwohl die Erörterung hierin in Zusammenhang mit dem Übertragen von Firmwareaktualisierungen erfolgt, können im Einklang mit den hierin offenbarten Lehren auch andere Arten von Software zu den Sensorknoten und/oder den IoT-Objekten übertragen werden.

[0014] Das Beispielsystem **100** weist eine erste Beispiel-Straßenlampe **102** aufweisend einen ersten Beispiel-Sensorknoten **104**, der mit dieser verbunden (z.B. darin eingebettet) ist, und eine zweite Beispiel-Straßenlampe **106** aufweisend einen zweiten Beispiel-Sensorknoten **108**, der mit jener verbunden ist, auf. Das Beispielsystem **100** kann zusätzliche, jeder der Straßenlampen **102**, **106** zugeordnete Sensorknoten und/oder zusätzliche Sensorknoten aufwei-

sende Straßenlampen aufweisen. Das Beispielsystem **100** von **Fig. 1** kann auch zusätzliche IoT-Objekte (z.B. Stoppschilder, Wasserzähler, Gebäude), welche Sensorknoten aufweisen, aufweisen.

[0015] Im Beispiel von **Fig. 1** weist die erste Straßenlampe **102** eine erste Beispiel-Kennung **105** auf und die zweite Straßenlampe **106** weist eine zweite Beispiel-Kennung **107** auf. Die Kennungen **105**, **107** können Markierungen (z.B. eine Nummer, einen Strichcode, eine Graphik) aufweisen, welche dazu verwendet werden, die Straßenlampen **102**, **106** (und somit die entsprechenden Sensorknoten **104**, **108**) voneinander zu unterscheiden. Wie hierin offenbart, werden die Kennungen **105**, **107** dazu verwendet, jene Straßenlampe **102**, **106**, welche den Sensorknoten **104** beziehungsweise **108** aufweist, zu identifizieren, welche die Firmwareaktualisierung empfangen soll. Zur einfacheren Erörterung kann die Übertragung einer Firmwareaktualisierung zu einem Sensorknoten in Verbindung mit dem ersten Sensorknoten **104** erörtert werden, wobei davon auszugehen ist, dass dieselbe oder eine ähnliche Beschreibung auch für den zweiten Sensorknoten **108** gilt.

[0016] Im Beispiel von **Fig. 1** weist der erste Sensorknoten **104** einen Beispielsensor **110** zum Sammeln von Daten, wie zum Beispiel Temperaturdaten für die Umgebung, in welcher die Straßenlampe **102** angeordnet ist, auf. Der erste Sensorknoten **104** weist einen Prozessor **112** (z.B. einen Mikrokontroller) auf. Der Sensor **110** überträgt Sensordaten (z.B. Temperaturdaten) zum Prozessor **112**, welche gespeichert, verarbeitet und/oder zu einem weiteren Knoten im Beispielsystem **100** übertragen werden sollen. Der Prozessor **112** steuert den Sensor **110** in Bezug auf die Daten, welche gesammelt werden, die Abtastrate, etc. Der Prozessor **112** kann eine oder mehrere Operationen an den Sensorsignaldaten durchführen, wie zum Beispiel Filtern der Rohsignaldaten, Entfernen von Rauschen aus den Signaldaten, Umwandeln der Signaldaten von analogen Daten in digitale Daten und/oder Analysieren der Daten. Der erste Sensorknoten **104** von **Fig. 1** weist auch eine Beispiel-Stromquelle **113**, wie zum Beispiel eine Batterie, auf.

[0017] Der erste Sensorknoten **104** weist ein Beispiel-Dualband-Funkfrequenz- (RF-) Modul **114** (z.B. einen RF-Senderempfänger) in Verbindung mit dem Prozessor **112** auf. In manchen Beispielen wird das Dualband-RF-Modul **114** durch den Prozessor **112** (z.B. einen Mikrokontroller) umgesetzt. Im Beispiel von **Fig. 1** kann das Dualband-RF-Modul **114** zwischen einem ersten Drahtlosmodus, wie zum Beispiel einem Energiespar-Drahtlosmodus (z.B. Kommunikation über das Sub-1-GHz-Band), und einem zweiten Drahtlosmodus, wie zum Beispiel einem Hochgeschwindigkeits-Drahtlosmodus (z.B. Kommunikation über das 2,4-GHz-Band und/oder das 5-GHz-Band), wechseln. In manchen Beispielen weist das

RF-Modul **114** zwei separate RF-Module auf, welche dem Energiespar-Drahtlosmodus und dem Hochgeschwindigkeits-Drahtlosmodus (z.B. ein erstes Sub-1-GHz-RF-Modul und ein zweites 2,4-GHz-RF-Modul) entsprechen.

[0018] Das RF-Modul **114** überträgt Daten, welche durch den Sensor **110** erzeugt werden, basierend auf einer oder mehreren durch den Prozessor **112** erzeugten Anweisungen zu einem Beispiel-Master-Knoten **116** in Kommunikation (z.B. Drahtloskommunikation) mit dem ersten Sensorknoten **104**. In manchen Beispielen kommuniziert das RF-Modul **114** mit dem Master-Knoten **116** im Energiespar-Drahtlosmodus, um Energieverbrauch durch den ersten Sensorknoten **104** zu verringern. In einigen solchen Beispielen kommuniziert der erste Sensorknoten **104** mit dem Master-Knoten **116** über das RF-Modul **114** unter Verwendung eines oder mehrere Kommunikationsprotokolle, wie zum Beispiel Zigbee™, SigFox™, etc. In manchen Beispielen weist der Prozessor **112** des ersten Sensorknotens **104** zwei oder mehr Prozessoren auf. In einigen solchen Beispielen ist mindestens einer der Prozessoren dem Steuern des Sensors **110** und/oder dem Verarbeiten der vom Sensor **110** empfangenen Daten gewidmet, und ein weiterer der Prozessoren steuert die Übertragung der Daten zum Beispiel über das RF-Modul **114** zum Master-Knoten **116**.

[0019] Im Beispiel von **Fig. 1** ist der Beispiel-Master-Knoten **116** von **Fig. 1** kommunikativ mit dem zweiten Sensorknoten **108** verbunden. Somit bilden der Master-Knoten **116**, der erste Sensorknoten **104** und der zweite Sensorknoten **108** ein lokales Sensornetzwerk **118** (z.B. ein Maschen-Netzwerk, ein Drahtlos-Netzwerk mit niedriger Geschwindigkeit). Der Master-Knoten **116** kann zum Beispiel eine Datenbank zum Speichern von vom ersten und/oder vom zweiten Sensorknoten, **104**, **108**, empfangenen Daten, und einen Prozessor zum Senden von Anweisungen zum ersten und/oder zum zweiten Sensorknoten **104**, **108**, aufweisen.

[0020] Im Beispiel von **Fig. 1** ist der Master-Knoten **116** kommunikativ (z.B. über Drahtlosverbindung(en)) mit einer oder mehreren cloudbasierten Beispiel-Vorrichtungen **120** (z.B. einer oder mehreren virtuellen Maschinen, Servern und/oder Prozessoren) verbunden. Im Beispiel von **Fig. 1** kommuniziert/kommunizieren die cloudbasierte(n) Vorrichtung(en) **120** mit einem/einer oder mehreren anderen Objekten oder Vorrichtungen. Wie in **Fig. 1** dargestellt, kann die cloudbasierte Vorrichtung **120** zum Beispiel über Wi-Fi zum Beispiel mit einem oder mehreren aus der Gruppe umfassend einen lokalen Server **122** und eine Benutzervorrichtung **124** (z.B. ein Smartphone, einen Personalcomputer, ein Tablet) kommunizieren. Im Beispiel von **Fig. 1** bilden die Straßenlampen **102**, **106** aufweisend die jeweiligen Sensorknoten **104**,

108, der Master-Knoten **116**, der lokale Server **122**, die Benutzervorrichtung **124** und die cloudbasierte(n) Vorrichtung(en) **120** ein Beispiel-IoT-Netzwerk **126**.

[0021] Im Beispiel von **Fig. 1** erfordert der erste Sensorknoten **104** zeitweise (z.B. in regelmäßigen und/oder unregelmäßigen Abständen, basierend auf einem oder mehreren Ereignissen, etc.) Firmwareaktualisierungen für Wartungszwecke und/oder zum Installieren neuer Merkmale/Funktionen, welche zum Beispiel durch den Prozessor **112** umzusetzen sind. Wie hierin offenbart, arbeitet der erste Sensorknoten **104** im Energiespar-Drahtlosmodus, indem er mittels des RF-Moduls **114**, zum Beispiel über das Sub-1-GHz-Frequenzband, mit dem Master-Knoten **116** und/oder dem zweiten Sensorknoten **108** kommuniziert. Wie ebenfalls hierin offenbart, kann das Übertragen einer FOTA-Aktualisierung zum ersten Sensorknoten **104** über den Master-Knoten **116**, wenn der erste Sensorknoten **104** im Energiespar-Drahtlosmodus arbeitet, zeitaufwändig sein, und in manchen Fällen aufgrund der Datenmenge, welche in Zusammenhang mit der Firmwareaktualisierung zu übertragen ist, das lokale Sensornetzwerk **118** destabilisieren. Außerdem kann das Dualband-RF-Modul **114** zwar in den Hochgeschwindigkeits-Drahtlosmodus umschalten, um zum Beispiel eine Kommunikation über das 2,4-GHz-WiFi-Frequenzband oder BLE zu ermöglichen, jedoch ist der erste Sensorknoten **104** in manchen Beispielen in einem Abstand vom Master-Knoten **116** angeordnet, welcher den Abstandsbereich, über welchen der Hochgeschwindigkeits-Drahtlosmodus funktioniert, übersteigt (z.B. mehr als 10 - 20 m vom Master-Knoten **116**).

[0022] Im Beispiel von **Fig. 1** wird ein unbemanntes Beispiel-Fluggerät (UAV) **128** eingesetzt, um eine FOTA-Aktualisierung am ersten Sensorknoten **104** bereitzustellen. Das UAV **128** von **Fig. 1** empfängt eine Anweisung von einem Beispiel-UAV-Manager **130**, zum ersten Sensorknoten **104** zu fliegen, und dem ersten Sensorknoten **104** die Firmwareaktualisierung zu übermitteln. Im dargestellten Beispiel ist der UAV-Manager **130** durch Software, die auf einem Prozessor **132** der Benutzervorrichtung **124**, einem Prozessor des lokalen Servers **122** und/oder durch die cloudbasierte(n) Vorrichtung(en) **120** ausgeführt wird, umgesetzt. In manchen Beispielen sind eine oder mehrere Komponenten des Beispiel-UAV-Managers **130** durch den Prozessor **132** der Benutzervorrichtung **124** umgesetzt, und eine oder mehrere andere Komponenten sind durch den lokalen Server **122** und/oder die cloudbasierte(n) Vorrichtung(en) **120** umgesetzt. Der UAV-Manager **130** kommuniziert über eine oder mehrere verdrahtete und/oder drahtlose Verbindungen mit dem UAV **128**. Wie in **Fig. 1** gezeigt, kann die Kommunikation zwischen dem UAV-Manager **130** und dem UAV **128** direkt (z.B. steht der Prozessor **132** der Benutzervorrichtung **124** in direkter Kommunikation mit dem UAV

128) oder indirekt (z.B. kommuniziert der lokale Server **122** mit dem UAV **128** über die Cloud-Computing-Umgebung) erfolgen.

[0023] Die Firmwareaktualisierung kann zum Beispiel an der Benutzervorrichtung **124** erzeugt und vom UAV-Manager **130** gespeichert werden. Der UAV-Manager **130** überträgt die Firmwareaktualisierung zum UAV **128** zur Speicherung auf dem UAV **128** (z.B. über verdrahtete und/oder drahtlose Verbindung(en)). Somit trägt das UAV **128** während des Flugs die Firmwareaktualisierung.

[0024] Im Beispielsystem **100** von **Fig. 1** erzeugt der UAV-Manager **130** eine oder mehrere Anweisungen für das UAV **128**, die Firmwareaktualisierung zum ersten Sensorknoten **104** zu übertragen. Zum Beispiel weist der UAV-Manager **130** das UAV **128** an, die Firmwareaktualisierung basierend auf den an der Benutzervorrichtung **124** empfangenen Benutzereingaben, welche anzeigen, dass die Firmwareaktualisierung einzuleiten ist, zu übermitteln. In anderen Beispielen weist der UAV-Manager **130** das UAV **128** automatisch an, die Firmwareaktualisierung basierend auf einem Sensorknoten-Aktualisierungszeitplan, welcher zum Beispiel durch den lokalen Server **122** und/oder die cloudbasierte(n) Vorrichtung(en) **120** umgesetzt wird, zu übermitteln.

[0025] Der Beispiel-UAV-Manager **130** von **Fig. 1** erzeugt eine Karte oder eine Wegpunktroute für das UAV **128** zur Erreichung von IoT-Objekten im IoT-Netzwerk, welche zu aktualisierende Sensorknoten aufweisen (z.B. die erste Straßenlampe **102**, welche den ersten Sensorknoten **104** aufweist), und übermittelt die Karte zum UAV **128**. Die vom UAV-Manager **130** erstellte IoT-Objektkarte legt einen Pfad für das UAV **128** fest, welchem dieses zu folgen hat, um einen oder mehrere Wegpunkte im IoT-Netzwerk **126** zu erreichen, die Karte aufweisend die erste Straßenlampe **102** und/oder die zweite Straßenlampe **106** und somit die jeweiligen Sensorknoten **104**, **108**. In manchen Beispielen wird die IoT-Karte vom UAV **128** basierend auf Benutzereingaben, welche über den Prozessor **132** der Benutzervorrichtung **124** empfangen werden, erstellt. Die Benutzereingaben können zum Beispiel eine Position des IoT-Objekts, welche den zu aktualisierenden Sensorknoten aufweist (z.B. eine Position der ersten Straßenlampe **102**), eine bevorzugte Flugroute für das UAV **128** (z.B. basierend auf zeitlichen Beschränkungen, bekannten Hindernissen, Windbedingungen) und die dem/den IoT-Objekt(en), welche den/die zu aktualisierende(n) Sensorknoten aufweisen, zugewiesenen Kennungen (z.B. die Kennung **105** der ersten Straßenlampe **102**), aufweisen. In anderen Beispielen erstellt der UAV-Manager **130** (z.B. am lokalen Server **122** und/oder der/den cloudbasierten Vorrichtung(en) **120**) automatisch die IoT-Objektkarte basierend auf zuvor vom UAV-Manager **130** gespeicherten Daten,

zum Beispiel in Bezug auf die Position der ersten Straßenlampe **102**, die Kennung **105** der ersten Straßenlampe **102**, etc.

[0026] Das Beispiel-UAV **128** weist einen Beispielprozessor **134** auf. Der Prozessor **134** des UAV **128** führt Software aus, um einen Beispiel-Sensorknotenaktualisierungsmanager **136** zum Übermitteln von Firmwareaktualisierung(en) zu einem oder mehreren Sensorknoten im IoT-Netzwerk **126** (z.B. dem ersten Sensorknoten **104**, dem zweiten Sensorknoten **108**) umzusetzen. Wie hierin offenbart, erzeugt der Sensorknotenaktualisierungsmanager **136** Anweisungen für den/die Sensorknoten zum Installieren der Firmwareaktualisierung(en) am/an den Sensorknoten.

[0027] Das Beispiel-UAV **128** weist ein erstes Beispiel- (z.B. Hochgeschwindigkeits-) RF-Modul **138** auf, welches WiFi- oder BLE-Kommunikation über eine hohe Frequenz (z.B. 2,4 GHz, 5 GHz, etc.) bereitstellt. Das Beispiel-UAV **128** weist ein zweites Beispiel-(z.B. Niedriggeschwindigkeits-) RF-Modul **140** auf, welches Kommunikation über große Entfernungen zum Beispiel über eine niedrige Frequenz (z.B. Sub-1-GHz) und/oder über Kommunikationsprotokolle, wie zum Beispiel Sigfox™, Zigbee™, etc., bereitstellt.

[0028] Das Beispiel-UAV **128** weist eine oder mehrere damit verbundene Beispielkameras **142** auf. Die Beispielkamera **142** von **Fig. 1** weist einen oder mehrere Sensoren (z.B. einen Rot-Grün-Blau- (RGB-) Sensor) auf, um Farbe, Licht, etc. zu erfassen, und um Bilddaten (z.B. Bildpunkte) zu erzeugen. Die Beispielkamera(s) **142** von **Fig. 1** weist/weisen auch mindestens einen Tiefensensor auf, um eine Entfernung der Kamera(s) **142** zum Beispiel vom Boden und/oder von einem Objekt im IoT-Netzwerk **126** (z.B. der ersten Straßenlampe **102**) zu messen, und um Tiefendaten zu erzeugen. In manchen Beispielen misst der/messen die Tiefensensor(en) die Tiefe mittels Projektion eines nahen Infrarotlichts durch einen oder mehrere Infrarotlaserprojektoren der Kamera(s) **142**, um 3D-Bilder zu erzeugen. Im Beispiel von **Fig. 1** wird/werden die Kamera(s) **142** durch RealSense™-Kameras umgesetzt, welche im Handel von der Intel™ Corporation erhältlich sind.

[0029] Im Beispiel von **Fig. 1** überträgt der UAV-Manager **130** die Firmwareaktualisierung und/oder die IoT-Objektkarte zum Sensorknotenaktualisierungsmanager **136**. Der Sensorknotenaktualisierungsmanager **136** weist das UAV **128** basierend auf der Karte (z.B. über von der am UAV **128** installierten Autopilotsoftware ausgeführte Anweisungen) an, zu einem Bereich zum Beispiel in der Nähe der ersten Straßenlampe **102** zu fliegen. Wenn sich das UAV **128** innerhalb eines Grenzabstands von der ersten Straßenlampe **102** befindet (z.B. basierend auf der Karte), erzeugt/erzeugen die Kamera(s) **142** Bilddaten in Be-

zug auf die Umgebung, in welcher das UAV **128** fliegt. In manchen Beispielen erzeugt/erzeugen die Kamera(s) **142** Tiefendaten in Bezug auf eine Tiefe des UAV **128** von einem oder mehreren Objekten im IoT-Netzwerk **126**.

[0030] Der Sensorknotenaktualisierungsmanager **136** verarbeitet die von der/den Kamera(s) erzeugten Bilddaten und/oder Tiefendaten, um zu ermitteln, ob das UAV **128** bei der ersten Straßenlampe **102** angelangt ist. Insbesondere wendet der Sensorknotenaktualisierungsmanager **136** eine oder mehrere Objekterkennungsregeln an, um festzustellen, ob das UAV **128** bei der ersten Straßenlampe **102** angelangt ist, basierend darauf, ob die Kennung **105** der ersten Straßenlampe **102** in den durch die Kamera(s) **142** erzeugten Bilddaten dargestellt wird. Wenn im Beispiel von **Fig. 1** der Sensorknotenaktualisierungsmanager **136** feststellt, dass sich das UAV **128** in der Nähe der ersten Straßenlampe **102** befindet, erzeugt der Sensorknotenaktualisierungsmanager **136** basierend auf den durch die Kamera(s) **142** erzeugten Bilddaten und Tiefendaten Anweisungen, um das UAV **128** in der Nähe (z.B. näher an) der ersten Straßenlampe **102** und/oder des Bereichs der ersten Straßenlampe **102**, mit welcher der erste Sensorknoten **104** verbunden (z.B. eingebettet) ist, zu positionieren. In manchen Beispielen werden die durch den Sensorknotenaktualisierungsmanager **136** erzeugten Anweisungen in Verbindung mit der Autopilotsoftware des UAV **128** erzeugt, um das UAV **128** zu navigieren. In manchen derartigen Beispielen schwebt das UAV **128** rund um den ersten Sensorknoten **104** und/oder einen Bereich der ersten Straßenlampe **102**, welche den ersten Sensorknoten **104** aufweist.

[0031] Wie oben offenbart, arbeitet der erste Sensorknoten **104** im Energiespar-Drahtlosmodus, während der Sensor **110** Daten sammelt, und die Sensordaten über das lokale Sensornetzwerk **118** (z.B. über das Sub-1-GHz-Frequenzband) zum Master-Knoten **116** übermittelt werden. Wie ebenfalls oben offenbart, stellt das zweite RF-Modul **140** des UAV **128** eine Niederfrequenzkommunikation, zum Beispiel über das Sub-1-GHz-Frequenzband, bereit. Somit kann das UAV **128** über Signale, welche durch das zweite RF-Modul **140** übertragen werden, mit dem ersten Sensorknoten **104** kommunizieren, wenn der erste Sensorknoten **104** im Energiespar-Drahtlosmodus arbeitet. In manchen Beispielen ist das UAV **128** derart gestaltet, dass es zur Kommunikation mit dem ersten Sensorknoten **104** anderen Netzwerksicherheitsparametern in Zusammenhang mit dem ersten Sensorknoten **104** und/oder dem lokalen Sensornetzwerk **118** (z.B. Authentifizieren, Schlüsselaustausch) entspricht.

[0032] Wenn das UAV **128** im Beispiel von **Fig. 1** in der Nähe der ersten Straßenlampe **102** positioniert

ist (z.B. nahe derselben schwebt), übermittelt der Sensorknotenaktualisierungsmanager **136** über das zweite RF-Modul **140** eine Mitteilung zum Prozessor **112** des ersten Sensorknotens **104**, welche Anweisungen für das Dualband-RF-Modul **114** aufweist, vom Energiespar-Drahtlosmodus (z.B. der Kommunikation über das niedrige Sub-1-GHz-Frequenzband) auf den Hochgeschwindigkeits-Drahtlosmodus (z.B. der Kommunikation über das hohe 2,4-GHz-Frequenzband) umzuschalten. Basierend auf den Anweisungen vom Sensorknotenaktualisierungsmanager **136** schaltet das Dualband-RF-Modul **114** in den Hochgeschwindigkeits-Drahtlosmodus um.

[0033] Wenn das Dualband-RF-Modul **114** des ersten Sensorknotens **104** auf den Hochgeschwindigkeits-Drahtlosmodus umschaltet, richtet der Sensorknotenaktualisierungsmanager **136** des UAV **128** über das erste RF-Modul **138** eine Punkt-zu-Punkt-Verbindung mit dem Prozessor **112** des ersten Sensorknotens **104** des UAV **128** ein. Wie hierin offenbart, stellt das erste RF-Modul **138** des UAV **128** eine Kommunikation über ein Hochfrequenzband, wie zum Beispiel das 2,4-GHz- oder das 5GHz-Frequenzband, bereit. Der Sensorknotenaktualisierungsmanager **136** übermittelt die FOTA-Aktualisierung über das erste RF-Modul **138** zum Prozessor **112** des ersten Sensorknotens **104**, wodurch er eine Hochgeschwindigkeits-Drahtlosübertragung der Firmwareaktualisierung zum ersten Sensorknoten bereitstellt.

[0034] Nachdem die FOTA-Aktualisierung dem ersten Sensorknoten **104** übermittelt worden ist, startet der erste Sensorknoten **104** in manchen Beispielen neu, um die Firmwareaktualisierung abzuschließen, und sendet eine Mitteilung an den Sensorknotenaktualisierungsmanager **136** des UAV **128**, um zu bestätigen, dass die FOTA-Aktualisierung erfolgreich war. In derartigen Beispielen arbeitet der erste Sensorknoten **104** nach wie vor im Hochgeschwindigkeits-Drahtlosmodus. Wenn der Sensorknotenaktualisierungsmanager **136** des UAV **128** die Bestätigungsmittteilung vom ersten Sensorknoten **104** empfängt, sendet der Sensorknotenaktualisierungsmanager **136** eine Mitteilung zum Prozessor **112** des ersten Sensorknotens **104**, welche Anweisungen für das Dualband-RF-Modul **114** aufweist, wieder zurück in den Energiespar-Drahtlosmodus (z.B. die Kommunikation über das niedrige Sub-1-GHz-Frequenzband) zu schalten. In manchen Beispielen sendet der Sensorknotenaktualisierungsmanager **136** während der und/oder nach dem Abschluss der FOTA-Aktualisierung (eine) Statusmitteilung(en) an den UAV-Manager **130** (z.B. über WiFi, Bluetooth). Die Statusmitteilung kann zum Beispiel einen Benutzer der Benutzervorrichtung **124** über den Status der Firmwareaktualisierung am ersten Sensorknoten **104** informieren.

[0035] Nach dem Abschluss der Firmwareaktualisierung am ersten Sensorknoten **104** weist der Sensorknotenaktualisierungsmanager **136** in manchen Beispielen das UAV **128** an, zu einem weiteren IoT-Objekt (z.B. der zweiten Straßenlampe **106**) zu fliegen, um eine FOTA-Sensorknotenaktualisierung basierend auf der IoT-Objektkarte zu übermitteln. Falls es keine weiteren zu aktualisierenden Sensorknoten gibt, weist der Sensorknotenaktualisierungsmanager **136** in manchen Beispielen das UAV **128** an, zu einer UAV-Lagerbasis zurückzukehren, bis das UAV **128** erneut eingesetzt werden soll.

[0036] Fig. 2 ist ein Blockdiagramm einer Beispielumsetzung des UAV-Managers **130** von Fig. 1. Wie oben erwähnt, ist der Beispiel-UAV-Manager **130** derart gestaltet, dass er ein oder mehrere UAV (z.B. das UAV **128** von Fig. 1) anweist, eine Firmwareaktualisierung an einen oder mehrere Sensorknoten (z.B. den ersten Sensorknoten **104** von Fig. 1) in einem IoT-Netzwerk (z.B. dem IoT-Netzwerk **126** von Fig. 1) zu übermitteln. Im Beispiel von Fig. 2 ist der UAV-Manager **130** durch einen oder mehrere der Prozessoren **132** der Benutzervorrichtung **124**, des lokalen Servers **122** und/oder der cloudbasierten Vorrichtung/en **120** (z.B. die Server, Prozessoren und/oder virtuellen Maschinen **120** von Fig. 1) umgesetzt. In manchen Beispielen sind eine oder mehrere Komponenten des UAV-Managers **130** in einer Cloud-Computing-Umgebung umgesetzt, und eine oder mehrere der Komponenten des UAV-Managers **130** sind durch den Prozessor **132** der Benutzervorrichtung **124** und/oder des lokalen Servers **122** umgesetzt. Im Beispiel von Fig. 2 steht der UAV-Manager **130** in Kommunikation mit dem Beispiel-Sensorknotenaktualisierungsmanager **136** des UAV **128** von Fig. 1 (z.B. über WiFi, Bluetooth). In manchen anderen Beispielen können eine oder mehrere der Komponenten des UAV-Managers **130** durch den Prozessor **134** des UAV **128** von Fig. 1 umgesetzt sein.

[0037] Der Beispiel-UAV-Manager **130** von Fig. 2 weist eine Beispieldatenbank **200** auf. In anderen Beispielen ist die Datenbank **200** außerhalb des UAV-Managers **130** an einer Position, auf welche der UAV-Manager **130** Zugriff hat, angeordnet. Wie oben offenbart, empfängt der UAV-Manager **130** in manchen Beispielen Beispiel-Firmwareaktualisierung(en) **202**, welche am/an den Sensorknoten zu installieren sind, von einem oder mehreren Prozessoren, welche zur Erstellung der Firmwareaktualisierung(en) **202** verwendet wurden. Der UAV-Manager **130** übermittelt Firmwareaktualisierung(en) **202** zum Sensorknotenaktualisierungsmanager **136** des UAV **128**. Die Beispieldatenbank **200** weist Mittel zum Speichern der Firmwareaktualisierung(en) **202** auf.

[0038] Der Beispiel-UAV-Manager **130** weist einen Beispielkommunikator **204** auf. Im dargestellten Beispiel stellt der Kommunikator **204** Mittel zum Kommu-

nizieren mit dem Sensorknotenaktualisierungsmanager **136** des UAV **128** bereit. Zum Beispiel übermittelt der Kommunikator **204** die Firmwareaktualisierung(en) **202** zum Speichern auf dem Sensorknotenaktualisierungsmanager **136**.

[0039] Der Beispiel-UAV-Manager **130** weist einen Beispiel-Firmwareaktualisierungsmanager **206** auf. Im dargestellten Beispiel stellt der Firmwareaktualisierungsmanager **206** Mittel zum Erkennen des/der Sensorknoten in einem IoT-Netzwerk, welche Firmwareaktualisierung(en) erhalten soll(en), bereit. In manchen Beispielen stellt der Firmwareaktualisierungsmanager **206** Mittel zur automatischen Bestimmung, ob der/die Sensorknoten zu aktualisieren sind, bereit.

[0040] In manchen Beispielen wendet der Beispiel-Firmwareaktualisierungsmanager **206** eine oder mehrere Beispiel-Firmwareaktualisierungsregeln **208** an, um zu bestimmen, ob der/die Sensorknoten **104**, **108** zu aktualisieren sind. Die Firmwareaktualisierungsregel(n) **208** kann/können durch eine oder mehrere Benutzereingaben definiert werden. Die Firmwareaktualisierungsregeln(n) **208** kann/können zum Beispiel Regeln darüber aufweisen, welche/r Sensorknoten die Aktualisierung(en) **202** erhalten soll(en), einen Zeitplan, welcher definiert, wann die Firmwareaktualisierung(en) **202** dem/den Sensorknoten bereitzustellen ist/sind, wie die Aktualisierung(en) **202** übermittelt werden soll(en) (z.B. als eine Aktualisierung oder in zwei oder mehr Paketen), etc. In manchen derartigen Beispielen bestimmt der Firmwareaktualisierungsmanager **206** automatisch, welche/r Sensorknoten basierend auf den Regeln **208** zu aktualisieren ist/sind. In anderen Beispielen definiert ein Benutzer den/die zu aktualisierenden Sensorknoten und die Art und Weise, auf welche die Aktualisierung(en) bereitzustellen ist/sind, über Benutzereingaben, welche zum Beispiel an der Benutzervorrichtung **124** empfangen werden. In derartigen Beispielen werden die Benutzereingaben durch den Firmwareaktualisierungsmanager **206** verarbeitet, um den/die zu aktualisierenden Sensorknoten zu ermitteln.

[0041] Basierend auf der Bestimmung des/der zu aktualisierenden Sensorknoten/s erzeugt der Firmwareaktualisierungsmanager **206** Beispiel-IoT-Objektidentifikationsdaten **209**. Wie oben offenbart, weist jedes der Objekte im Beispiel-IoT-Netzwerk **126** eine Kennung auf, wie zum Beispiel die erste Kennung **105**, welche der ersten Straßenlampe **102** zugewiesen ist, und die zweite Kennung **107**, welche der zweiten Straßenlampe **106** zugewiesen ist. Wie ebenfalls oben offenbart, bestimmt der Beispiel-Sensorknotenaktualisierungsmanager **136**, basierend auf der dem Objekt zugewiesenen Kennung, dass das UAV **128** an dem IoT-Objekt angelangt ist,

welches den/die zu aktualisierenden Sensorknoten aufweist.

[0042] In manchen Beispielen erzeugt der Firmwareaktualisierungsmanager **206** die IoT-Objektidentifikationsdaten **209** basierend auf Benutzereingaben, welche anzeigen, dass die betreffenden IoT-Objekt-sensorknoten zu aktualisieren sind. In manchen Beispielen erzeugt der Firmwareaktualisierungsmanager **206** automatisch die IoT-Objektidentifikationsdaten **209** basierend auf zuvor gespeicherten Identifikationsdaten als Reaktion auf die automatische Ermittlung, dass bestimmte Sensorknoten zu aktualisieren sind (z.B. basierend auf den Firmwareaktualisierungsregeln **208**).

[0043] Die IoT-Objektidentifikationsdaten **209** weisen zum Beispiel eine Zuordnung von Kennungen für ein oder mehrere Objekte im IoT-Netzwerk auf, welches den/die über das UAV **128** von Fig. 1 zu aktualisierenden Sensorknoten aufweist. In manchen Beispielen weisen die IoT-Objektidentifikationsdaten **209** eine Auflistung der Kennungen mindestens eines der Abschnitte des/der IoT-Objekts/Objekte im IoT-Netzwerk auf. In derartigen Beispielen weisen die IoT-Objektidentifikationsdaten **209** Flags (Markierungen) für das/die IoT-Objekt/e auf, welche/s (den) zu aktualisierende/n Sensorknoten aufweist/aufweisen. Die Beispieldatenbank **200** speichert die IoT-Objektidentifikationsdaten **209**.

[0044] In manchen Beispielen erzeugt der Firmwareaktualisierungsmanager **206** Beispiel-Firmwareaktualisierungsanweisungen **211** basierend auf (einer) Bedienereingabe(n) und/oder der/den Firmwareaktualisierungsregel(n) **208**. Die Firmwareaktualisierungsanweisung(en) **211** werden durch den Kommunikator **204** zum Sensorknotenaktualisierungsmanager **136** des UAV **128** übermittelt. Die Anweisung(en) **211** kann/können zum Beispiel Anweisungen für das UAV **128**, zu einer bestimmten Zeit basierend auf einem vorher definierten Firmwareaktualisierungsübermittlungszeitplan zum/zu den Sensorknoten zu fliegen, Anweisungen für die Firmwareaktualisierung(en) **202**, in zwei oder mehr Paketen übermittelt zu werden, etc., aufweisen.

[0045] Der Beispiel-UAV-Manager **130** weist einen Beispielkartengenerator **210** auf. Im dargestellten Beispiel stellt der Kartengenerator **210** Mittel zum Erzeugen einer oder mehrerer Beispiel-IoT-Objektkarten **212** bereit, welche eine Route definieren, der das UAV **128** folgen muss, um das/die IoT-Objekt/e, welche/s den/die zu aktualisierenden Sensorknoten aufweist/aufweisen, zu erreichen. In Beispielen, in welchen der Firmwareaktualisierungsmanager **206** automatisch ermittelt, dass der/die Sensorknoten zu aktualisieren sind, (z.B. basierend auf den Firmwareaktualisierungsregeln **208**), erzeugt der Kartengenerator **210** automatisch die IoT-Objektkarte(n) **212**. In an-

deren Beispielen erstellt der Kartengenerator **210** die Karte/n **212** als Reaktion auf das Empfangen einer Benutzereingabe, welche den Kartengenerator **210** anweist, die Karte/n zu erstellen.

[0046] Der Beispiel-Kartengenerator **210** erstellt die Karten basierend auf Beispiel-Zuordnungsregeln **214** und/oder Beispiel-IoT-Objektpositionsdaten **216**, welche in der Datenbank **200** gespeichert sind. In **Fig. 2** können die Zuordnungsregeln **214** und die IoT-Objektpositionsdaten **216** durch Benutzereingaben definiert werden. Die Zuordnungsregeln **214** können zum Beispiel Benutzerpräferenzen für die zu erstellende(n) Route(n), welchen das/die UAV **128** zu folgen hat/haben, basierend auf Zeit, Hindernissen, welchen das/die UAV **128** ausweichen muss/müssen, Wetterbedingungen (z.B. Wind), etc., aufweisen. Die IoT-Objektpositionsdaten **216** weisen Positionsdaten über das/die IoT-Objekt/e, welchen jeder der Sensorknoten zugeordnet ist (z.B. die Positionen der jeweiligen Straßenlampen **102**, **106**), die Positionen der jeweiligen Sensorknoten in Bezug auf das/die IoT-Objekt/e (z.B. ist der erste Sensorknoten **104** in der Nähe eines Abschnitts der ersten Straßenlampe **102**, welcher das Licht aufweist, angeordnet), etc. auf.

[0047] Der Beispielkartengenerator **210** von **Fig. 2** erstellt die IoT-Objektkarte/n **212**, welche einen oder mehrere Wegpunkte entsprechend jedem zu aktualisierenden Sensorknoten aufweisen können. Eine Beispiel-IoT-Objektkarte **212** kann eine Route für den Flug von einer UAV-Basis zur ersten Straßenlampe **102** zum Aktualisieren des ersten Sensorknotens **104**, eine Route für den Flug von der ersten Straßenlampe **102** zur zweiten Straßenlampe **106** zum Aktualisieren des zweiten Sensorknotens **108**, eine Route für den Flug von der zweiten Straßenlampe **106** zu einem weiteren IoT-Objekt und/oder zur Rückkehr zur UAV-Basis, etc. aufweisen. Im Beispiel von **Fig. 2** ist/sind die Karte(n) **212** in der Datenbank **200** gespeichert. Die in der Datenbank **200** gespeicherte(n) Karte(n) **212** kann/können durch den UAV-Manager **130** abgerufen werden, wenn der/die Sensorknoten des/der IoT-Objekts/Objekte, welche(s) der/den Karte(n) **212** zugeordnet ist/sind, Firmwareaktualisierungen (z.B. an einem Zeitpunkt in der Zukunft) erfordert/erfordern, um eine Wirksamkeit des UAV-Managers **130** zur Bereitstellung von Anweisungen für das UAV **128** zu erhöhen. Im Beispiel von **Fig. 2** übermittelt der Kommunikator **204** die Karte(n) **212** zum Sensorknotenaktualisierungsmanager **136** des UAV **128**.

[0048] Wie hierin offenbart, erzeugt der Sensorknotenaktualisierungsmanager **136** des UAV **128** von **Fig. 1** in manchen Beispielen eine oder mehrere Beispielstatusmitteilungen **218** während der FOTA-Aktualisierung und/oder nach dem Abschluss der FOTA-Aktualisierung, und übermittelt (z.B. über Wifi, Bluetooth) die Mitteilung(en) **218** zum UAV-Manager **130**. Die Statusmitteilung(en) **218** können zum Bei-

spiel Mitteilungen aufweisen, dass die FOTA-Aktualisierung(en) im Gange ist/sind, dass die Firmwareaktualisierung(en) erfolgreich installiert worden ist/sind, ob Fehler auftraten, oder nicht, etc. Die vom Sensorknotenaktualisierungsmanager **136** empfangenen Statusmitteilung(en) **218** werden in der Datenbank **200** gespeichert.

[0049] Der Beispiel-UAV-Manager **130** weist einen Beispiel-UAV-Statusauswerter **220** auf. Der UAV-Statusauswerter **220** stellt Mittel zum Analysieren der Statusmitteilung(en) **218**, welche vom Sensorknotenaktualisierungsmanager **136** des UAV **128** von **Fig. 1** empfangen wird/werden, bereit. In manchen Beispielen stellt der UAV-Statusauswerter **220** Mittel zum Erzeugen von Anweisungen, welche über den Kommunikator **204** zum UAV **128** zu übermitteln sind, basierend auf der/den Statusmitteilung(en) **218** bereit. Falls die Statusmitteilung(en) **218** zum Beispiel anzeigt/anzeigen, dass ein Fehler während der FOTA-Aktualisierung aufgetreten ist, kann der UAV-Statusauswerter **220** Anweisungen zum Beenden der FOTA-Aktualisierung, zur Rückkehr des UAV **128** zur UAV-Basis, etc. erzeugen. Als weiteres Beispiel kann der UAV-Statusauswerter **220**, falls die Firmwareaktualisierung **202** in zwei oder mehr Paketen zu übermitteln ist, den Sensorknotenaktualisierungsmanager **136** des UAV **128** von **Fig. 1** anweisen, das zweite Paket zu übermitteln, nachdem der UAV-Statusauswerter **220** überprüft hat, dass eine Statusmitteilung **218** empfangen worden ist, welche bestätigt, dass die erste Paketaktualisierung erfolgreich war. In anderen derartigen Beispielen wartet der Sensorknotenaktualisierungsmanager **136** des UAV **128** nicht auf Anweisungen vom UAV-Statusauswerter, bevor er mit der Aktualisierung des zweiten Pakets beginnt. In manchen Beispielen erzeugt der UAV-Statusauswerter **220** Anweisungen zum Anzeigen der Statusmitteilung(en) **218** zum Beispiel über eine Anzeige der Benutzervorrichtung **124**.

[0050] Während in **Fig. 2** ein Beispiel der Umsetzung des Beispiel-UAV-Managers **130** dargestellt ist, können eines/einer/eine oder mehrere der Elemente, Prozesse und/oder Vorrichtungen, welche in **Fig. 2** dargestellt sind, kombiniert, geteilt, umgestellt, ausgelassen, weggelassen und/oder auf irgendeine andere Weise umgesetzt werden. Ferner können die Beispiel-Datenbank **200**, der Beispiel-Kommunikator **204**, der Beispiel-Firmwareaktualisierungsmanager **206**, der Beispiel-Kartengenerator **210**, der Beispiel-UAV-Statusauswerter **220** und/oder allgemeiner der Beispiel-UAV-Manager **130** von **Fig. 2** durch Hardware, Software, Firmware und/oder irgendeine Kombination von Hardware, Software und/oder Firmware umgesetzt werden. Somit könnten zum Beispiel die Beispiel-Datenbank **200**, der Beispiel-Kommunikator **204**, der Beispiel-Firmwareaktualisierungsmanager **206**, der Beispiel-Kartengenerator **210** und/oder der Beispiel-UAV-Statusauswer-

ter **220** und/oder allgemeiner der Beispiel-UAV-Manager **130** von **Fig. 2** durch eine oder mehrere analoge oder digitale Schaltungen, Logikschaltungen, programmierbare Prozessoren, anwendungsspezifische integrierte Schaltungen (ASIC(s)), programmierbare Logikvorrichtung(en) (PLD(s)) und/oder feldprogrammierbare Logikvorrichtung(en) (FPLD(s)) umgesetzt werden. Beim Lesen irgendeines der Vorrichtungs- oder Systemansprüche dieses Patents zur Abdeckung einer reinen Software- und/oder Firmwareumsetzung ist hiermit mindestens das Beispiel, die Beispiel-Datenbank **200**, der Beispiel-Kommunikator **204**, der Beispiel-Firmwareaktualisierungsmanager **206**, der Beispiel-Kartengenerator **210**, der Beispiel-UAV-Statusauswerter **220** und/oder allgemeiner der Beispiel UAV-Manager **130** von **Fig. 2** ausdrücklich derart definiert, dass er eine nichtflüchtige computerlesbare Speichervorrichtung oder Speicherplatte, wie zum Beispiel einen Speicher, eine DVD (digitale versatile Disk), eine CD (Kompaktdiskette), eine Blu-ray-Disk, etc., aufweist, welche die Software und/oder Firmware aufweist. Ferner kann der Beispiel-UAV-Manager **130** von **Fig. 1** und **Fig. 2** auch ein/eine/einen oder mehrere Elemente, Prozesse und/oder Vorrichtungen zusätzlich zu, oder anstelle von, jenen, die in **Fig. 1** und **Fig. 2** dargestellt sind, aufweisen, und/oder kann mehr als eines/einen/eine der oder sämtliche der dargestellten Elemente, Prozesse und Vorrichtungen aufweisen.

[0051] **Fig. 3** ist ein Blockdiagramm einer Beispielumsetzung des Sensorknotenaktualisierungsmanagers **136** von **Fig. 1**. Wie oben erwähnt ist der Beispiel-Sensorknotenaktualisierungsmanager **136** derart gestaltet, dass er eine oder mehrere Firmwareaktualisierung(en) (z.B. die Firmwareaktualisierung(en) **202** von **Fig. 2**) an einen oder mehrere Sensorknoten (z.B. die Sensorknoten **104**, **108** von **Fig. 1**) übermittelt. Im Beispiel von **Fig. 3** ist der Sensorknotenaktualisierungsmanager **136** durch den Prozessor **134** des UAV **128** umgesetzt.

[0052] Der Beispiel-Sensorknotenaktualisierungsmanager **136** weist eine Beispieldatenbank **300** auf. In anderen Beispielen ist die Datenbank **300** außerhalb des Sensorknotenaktualisierungsmanagers **136** an einer Position, auf welche der Manager **136** Zugriff hat, angeordnet. Wie oben offenbart wird/werden in manchen Beispielen die am/an den Sensorknoten zu installierenden Firmwareaktualisierung(en) **202** vom UAV-Manager **130** von **Fig. 1** und/oder 2 zum Sensorknotenaktualisierungsmanager **136** übertragen. In anderen Beispielen wird/werden die Firmwareaktualisierung(en) **202** von einem anderen Prozessor (z.B. dem/den Prozessor(en), an welchem/welchen die Firmwareaktualisierung(en) erstellt wurde(n) zum Sensorknotenaktualisierungsmanager **136** übermittelt. Die Datenbank **300** von **Fig. 3** weist Mittel zum Speichern der Firmwareaktualisierung(en) **202** auf.

[0053] Wie ebenfalls oben offenbart, empfängt der Sensorknotenaktualisierungsmanager **136** die IoT-Objektidentifikationsdaten **209**, welche durch den Beispiel-Firmwareaktualisierungsmanager **206** von **Fig. 2** erzeugt wurden und jene/s IoT-Objekt/e angeben, welche/s den/die zu aktualisierenden Sensorknoten aufweist/aufweisen. Die Datenbank **300** von **Fig. 3** stellt Mittel zum Speichern der IoT-Objektidentifikationsdaten **209** bereit. In manchen Beispielen empfängt der Sensorknotenaktualisierungsmanager **136** (eine) Firmwareaktualisierungsanweisung(en) **211**, welche durch den Firmwareaktualisierungsmanager **206** von **Fig. 2** erstellt worden ist/sind. Die Datenbank **300** von **Fig. 3** weist Mittel zum Speichern der Firmwareaktualisierungsanweisung(en) **211** auf.

[0054] Wie ebenfalls oben offenbart, empfängt der Sensorknotenaktualisierungsmanager **136** in manchen Beispielen die IoT-Objektkarte(n) **212** vom Kartengenerator **210** des UAV-Managers. Die Beispiel-Datenbank **300** von **Fig. 3** weist Mittel zum Speichern der Karte(n) **212** auf.

[0055] Der Beispiel-Sensorknotenaktualisierungsmanager **136** von **Fig. 3** weist einen Beispielnavigator **302** auf. Der Navigator **302** stellt Mittel zum Navigieren des UAV **128** zum/zu den IoT-Objekt(en), welche/s den/die zu aktualisierenden Sensorknoten aufweist, bereit. Im Beispiel von **Fig. 3** verarbeitet der Navigator **302** die IoT-Objektkarte(n) **212** und erzeugt Anweisungen für das UAV **128** für den Flug entlang einer/von Route(n), welche in der/den Karte(n) **212** festgelegt worden ist/sind. In manchen Beispielen werden die durch den Navigator **302** erzeugten Anweisungen in Verbindung mit der Autopilotsoftware, welche am UAV **128** installiert ist, ausgeführt.

[0056] Im Beispiel von **Fig. 3** erzeugt der Navigator **302** Beispiel-UAV-Positionsdaten **303**, während das UAV **128** fliegt. Die UAV-Positionsdaten **303** können zum Beispiel eine Position des UAV **128** relativ zu einem oder mehreren Wegpunkten in der/den Karte(n) **212** im Wesentlichen in Echtzeit aufweisen. Zum Beispiel können die UAV-Positionsdaten **303** eine Position des UAV **128** relativ zur ersten Straßenlampe **102** von **Fig. 1** aufweisen. Die UAV-Positionsdaten **303** werden in der Datenbank **300** gespeichert.

[0057] Der Beispiel-Sensorknotenaktualisierungsmanager **136** von **Fig. 3** weist einen Beispiellkamerananager **304** auf. Im dargestellten Beispiel stellt der Kameramanager **304** Mittel zum Steuern der Kamera(s) **142** des UAV **128** bereit. Der Beispiel-Kameramanager **304** von **Fig. 3** wendet eine oder mehrere Kameraregel(n) **305** an, um zum Beispiel zu steuern, wann die Kamera(s) **142** beginnen soll(en) Bild- und/oder Tiefendaten zu sammeln, eine Dauer, für welche die Kamera(s) **142** Daten sammeln soll(en), einen Typ der zu sammelnden Daten (z.B. Rot-Grün-Blau-Bilddaten), etc. In manchen Beispielen weist/weisen

die Kameraregel(n) **305** vordefinierte Grenzabstände relativ zu dem IoT-Objekt, welches den zu aktualisierenden Sensorknoten aufweist, auf, welche definieren, wann die Kamera(s) **142** beginnen soll(en), Daten zu sammeln. Die vordefinierten Grenzabstände können auf Benutzereingaben, der/den Karte(n) **212**, den UAV- Positionsdaten **303**, etc. basieren.

[0058] Der Beispiel-Kameramanager **304** erzeugt Beispiel-Kameraauslöseranweisungen **306**, um die Kamera(s) **142** zum Sammeln von Bild- und/oder Tiefendaten basierend auf den Kameraregeln **305** zu aktivieren. Der Beispiel-Sensorknotenaktualisierungsmanager **136** weist einen Beispiel-Kommunikator **308** auf, um die Kameraauslöseranweisungen **306** zur/zu den Kamera(s) **142** zu übermitteln. Als Reaktion auf die Anweisungen **306** erzeugt/erzeugen die Kamera(s) **142** von **Fig. 1** Beispiel-Bilddaten **310** (z.B. Farbdaten, Lichtdaten). In manchen Beispielen erzeugt/erzeugen die Kamera(s) **142** Beispiel-Tiefendaten **312**, welche Messungen des Abstands/der Abstände der Kamera(s) **142** zum Beispiel vom Boden und/oder von Objekten im IoT-Netzwerk **126** von **Fig. 1** (z.B. der ersten Straßenlampe **102**) aufweisen. Im Beispiel von **Fig. 3** werden die Bilddaten **310** und die Tiefendaten **312** in der Beispiel-Datenbank **300** gespeichert.

[0059] Der Beispiel-Kameramanager **304** von **Fig. 3** führt eine oder mehrere Datenverarbeitungstechniken an den Bilddaten **310** und/oder den Tiefendaten **312** aus. Zum Beispiel filtert der Kameramanager **304** die Bilddaten **310**, um die Helligkeit anzupassen (z.B. zu reduzieren). Der Kameramanager **304** richtet die Bilddaten **310** und die Tiefendaten **312** aus, um ausgerichtete Beispiel-Bilddaten **313** zu erstellen, welche 3D-Koordinaten enthalten (z.B. basierend auf den Bilddaten **310**, welche in der X-Y-Ebene gesammelt werden, und den Tiefendaten **312**, welche in der Z-Ebene gesammelt werden). Die ausgerichteten Bilddaten **313** werden in der Datenbank **300** von **Fig. 3** gespeichert.

[0060] Der Beispiel-Sensorknotenaktualisierungsmanager **136** weist einen Beispiel-IoT-Objektidentifikator **314** auf. Im dargestellten Beispiel stellt der IoT-Objektidentifikator **314** Mittel zum Identifizieren oder Erkennen des/der IoT-Objekt(e) im IoT-Netzwerk, welche(s) den/die zu aktualisierenden Sensorknoten aufweisen (z.B. die erste Straßenlampe **102**), während das UAV **128** zum Objekt fliegt, bereit. Im Beispiel von **Fig. 3** identifiziert oder erkennt der IoT-Objektidentifikator **314** das/die IoT-Objekt(e) basierend auf den IoT-Objektidentifikationsdaten **209** und einer oder mehreren Beispiel-IoT-Objektidentifikationsregel(n) **316**, welche in der Datenbank **300** gespeichert ist/sind. In manchen Beispielen erkennt der IoT-Objektidentifikator **314** das/die IoT-Objekt(e) basierend auf den UAV-Positionsdaten **303**, den Bild-

daten **310**, den Tiefendaten **312** und/oder den ausgerichteten Bilddaten **313**.

[0061] Die IoT-Objektidentifikationsregeln **316** können zum Beispiel Algorithmen, welche vom IoT-Objektidentifikator **314** umgesetzt werden sollen, um (1) ein oder mehrere IoT-Objekte von Interesse (z.B. eine Straßenlampe) im Vergleich zu anderen Objekten im IoT-Netzwerk (z.B. einem Stoppschild) zu unterscheiden und (2) um das/die bestimmte(n) IoT-Objekt(e), welche den/die zu aktualisierenden Sensorknoten aufweisen (z.B. die erste Straßenlampe **102**, welche den ersten Sensorknoten **104** aufweist) zu identifizieren, aufweisen. Die IoT-Identifikationsregeln oder Algorithmen können durch Benutzereingaben definiert werden.

[0062] Zum Beispiel vergleicht der IoT-Objektidentifikator **314** die Bilddaten **310** und/oder die ausgerichteten Bilddaten **313** mit Beispiel-Referenzbilddaten **320**, welche in der Datenbank **300** gespeichert sind, um zu ermitteln, ob das UAV **128** in der Nähe eines IoT-Objekts fliegt, welches im Wesentlichen gleich dem IoT-Objekt, welches den zu aktualisierenden Sensorknoten aufweist, ist. Zum Beispiel ermittelt der IoT-Objektidentifikator **314** basierend auf den Bilddaten **310**, **313** und den Referenzbilddaten **320**, ob das UAV **128** in der Nähe einer oder mehrerer Straßenlampen fliegt. In manchen Beispielen verwendet der IoT-Objektidentifikator **314** die durch den Navigator **302** und/oder die Karte(n) **212** erzeugten Positionsdaten **303**, um zu bestätigen, dass sich das UAV in einem Bereich befindet, in welchem Straßenlampen angeordnet sind. Die Referenzbilddaten **320** können zuvor durch die Kamera(s) **142** des UAV **128** erzeugt und in der Datenbank **300** (z.B. während früheren Flügen basierend auf der/den Karte(n) **212**) gespeicherte Bilddaten, über Benutzereingaben bereitgestellte Bilddaten, etc. aufweisen.

[0063] Wenn der IoT-Objektidentifikator **314** feststellt, dass sich das UAV **128** in der Nähe des/der IoT-Objekt(e) von Interesse (z.B. einer oder mehreren Straßenlampen) befindet, ermittelt der IoT-Objektidentifikator **314**, ob das UAV **128** das IoT-Objekt erreicht hat, welches den/die zu aktualisierenden Sensorknoten aufweist (z.B. die erste Straßenlampe, welche den ersten Sensorknoten **104** aufweist). Um das IoT-Objekt mit dem/den zu aktualisierenden Sensorknoten zu identifizieren, analysiert der IoT-Objektidentifikator **314** die Bilddaten **310** und/oder die ausgerichteten Bilddaten **313**, um die in den Bilddaten erfassten Kennung(en) des/der IoT-Objekts/Objekte zu entdecken. Der IoT-Objektidentifikator **314** vergleicht die in den Bilddaten **310**, **313** erfassten Kennung(en) des/der IoT-Objekt(e) mit den IoT-Objektidentifikationsdaten **209**, welche in der Datenbank **300** gespeichert sind. Basierend auf dem Vergleich der Kennungen bestimmt der IoT-Objektidentifikator **314**, ob sich das UAV **128** in der Nähe (z.B. innerhalb eines

Grenzabstands) des IoT-Objekts, welches den/die zu aktualisierenden Sensorknoten aufweist, befindet.

[0064] In manchen Beispielen stellt der Beispiel-IoT-Objektidentifikator **314** basierend auf dem Vergleich der Kennung(en) in den Bilddaten **310**, **313** mit den IoT-Objektidentifikationsdaten **209**, welche in der Datenbank **300** gespeichert sind, fest, dass sich das UAV **128** nicht in der Nähe des IoT-Objekts befindet, welches den/die zu aktualisierenden Sensorknoten aufweist. In derartigen Beispielen weist der Beispiel-Navigator **302** das UAV **128** an, weiterhin der Karte **212** zu folgen, um das betreffende IoT-Objekt (z.B. die erste Straßenlampe **102**) zu erreichen. Das Beispiel-UAV **128** folgt weiterhin der Karte **212**, bis der IoT-Objektidentifikator **314** (z.B. basierend auf dem Vergleich der Kennung(en) in den Bilddaten **310**, **313**, welche während des Flugs erzeugt werden, mit den IoT-Objektidentifikationsdaten **209**) feststellt, dass sich das UAV **128** in der Nähe des IoT-Objekts, welches den/die zu aktualisierenden Sensorknoten aufweist, befindet.

[0065] In manchen Beispielen stellt der IoT-Objektidentifikator **314** ausschließlich basierend auf der UAV-Positionsdaten **303** oder den UAV-Positionsdaten **303** und der/den Karte(n) **212** fest, dass sich das UAV **128** in der Nähe des IoT-Objekts, welches den/die zu aktualisierenden Sensorknoten aufweist, befindet. In manchen Beispielen bestimmt der IoT-Objektidentifikator **314** basierend auf einem Stärkepegel eines Signals, welches vom/von den Sensorknoten, welche/r aktualisiert werden soll(en), ausgesendet wird, und welches vom UAV **128** erfasst wird, wenn sich das UAV **128** in der Nähe des IoT-Objekts, welches den Sensorknoten aufweist, befindet, dass sich das UAV **128** in der Nähe des IoT-Objekts, welches den/die zu aktualisierenden Sensorknoten aufweist, befindet.

[0066] Wenn der IoT-Objektidentifikator **314** feststellt, dass sich das UAV **128** in der Nähe des IoT-Objekts, welches den/die zu aktualisierenden Sensorknoten aufweist, befindet, weist der Beispielnavigator **302** in manchen Beispielen das UAV **128** an, näher an das IoT-Objekt (z.B. die erste Straßenlampe **102**) heranzufiegen. In manchen Beispielen analysiert der Navigator **302** die ausgerichteten Bilddaten **313**, welche die Tiefenmessungen aufweisen, um das UAV **128** in der Nähe des IoT-Objekts und/oder einer Position des Sensorknotens relativ zum IoT-Objekt zu positionieren. Zum Beispiel kann der Navigator **302** das UAV **128** anweisen, nahe des Lichtabschnitts der ersten Straßenlampe **102** von **Fig. 1** zu schweben, wo der erste Sensorknoten **104** angeordnet sein kann, anstatt in Bodennähe zu schweben. Somit positioniert der Navigator **302** das UAV **128**, um die Kommunikation mit dem/den Sensorknoten zur Übermittlung der Firmwareaktualisierung(en) **202**

über ein oder mehrere Drahtlos-Aktualisierungen zu ermöglichen.

[0067] Der Beispiel-Sensorknotenaktualisierungsmanager **136** weist einen Beispiel-Sensorknotenverbindungsmanager **322** auf. Im dargestellten Beispiel stellt der Sensorknotenverbindungsmanager **322** Mittel zum Kommunizieren mit dem/den Sensorknoten, welche(r) die Firmwareaktualisierung(en) **202** über das UAV **128** erhalten soll(en), bereit. Wie oben offenbart, arbeitet ein zu aktualisierender Sensorknoten, wie zum Beispiel der erste Sensorknoten **104** von **Fig. 1**, typischerweise im Energiespar-Drahtlosmodus (z.B. Kommunikation über das Sub-1-GHz-Frequenzband), wenn er Sensordaten sammelt und die Sensordaten zum Master-Knoten **116** von **Fig. 1** übermittelt. Der Beispiel-Sensorknotenverbindungsmanager **322** von **Fig. 3** weist den Kommunikator **308** an, eine Kommunikation mit dem ersten Sensorknoten **104**, welcher im Energiespar-Drahtlosmodus arbeitet, einzurichten, indem er eine oder mehrere Mitteilungen über das zweite (z.B. Niedriggeschwindigkeits-) RF-Modul **140** des UAV **128** aussendet. Der Beispiel-Sensorknotenverbindungsmanager **322** weist den Kommunikator **308** an, ein oder mehrere Kommunikationsprotokolle zu verwenden, um mit dem ersten Sensorknoten **104** über das zweite RF-Modul **140** (z.B. WiFi über das Sub-1-GHz-Frequenzband, Zigbee™, SigFox™) zu kommunizieren. In manchen Beispielen weist der Sensorknotenverbindungsmanager **322** den Kommunikator **308** an, mit dem lokalen Sensornetzwerk **118** von **Fig. 1** zu kommunizieren, um die Einhaltung von Sicherheitsparametern, Authentifizierungsparametern, etc. durch das UAV **128** zu überprüfen und somit dem UAV **128** zu ermöglichen, mit den Sensorknoten im lokalen Sensornetzwerk **118** zu kommunizieren. Somit dient im Beispiel von **Fig. 3** das UAV **128** als ein Knoten (z.B. ein Hostknoten) im lokalen Sensornetzwerk **118**, um mit dem ersten Sensorknoten **104** zu kommunizieren.

[0068] Der Beispiel-Sensorknotenverbindungsmanager **322** von **Fig. 3** erstellt eine oder mehrere Beispiel-RF-Schaltauslösermitteilung(en) **324** für den zu aktualisierenden Sensorknoten. Die RF-Schaltauslösermitteilung **324** weist zum Beispiel Anweisungen für den ersten Sensorknoten **104** auf, vom ersten oder Energiespar-Drahtlosmodus in den zweiten oder Hochgeschwindigkeits-Drahtlosmodus (z.B. Kommunikation über das Hochfrequenz-2,4-GHz- oder 5-GHz-Frequenzband) umzuschalten. Der Beispielskommunikator **308** von **Fig. 3** übermittelt die RF-Schaltauslösermitteilung(en) **324** zum Beispiel über das Sub-1-GHz-Frequenzband zum ersten Sensorknoten **104**.

[0069] Wie oben offenbart, schaltet das Dualband-RF-Modul **114** des ersten Sensorknotens **104** von **Fig. 1** als Reaktion auf das Empfangen der RF-

Schaltauslösermitteilung(en) **324** vom Energiespar-Drahtlosmodus in den Hochgeschwindigkeits-Drahtlosmodus um. In manchen Beispielen sendet der erste Sensorknoten **104** eine Mitteilung an den Sensorknotenverbindungsmanager **322**, welche bestätigt, dass der erste Sensorknoten **104** im Hochgeschwindigkeits-Drahtlosmodus arbeitet. Wenn der erste Sensorknoten **104** im Hochgeschwindigkeits-Drahtlosmodus arbeitet, weist der Sensorknotenverbindungsmanager **322** den Kommunikator **308** an, zum Beispiel über das 2,4-GHz-Frequenzband durch das erste (z.B. Hochgeschwindigkeits-) RF-Modul **138** eine Punkt-zu-Punkt-Verbindung mit dem ersten Sensorknoten **104** einzurichten. Somit ist die Kommunikation zwischen dem Sensorknotenaktualisierungsmanager **136** des UAV **128** und dem ersten Sensorknoten **104** eingerichtet, wenn der erste Sensorknoten **104** im Hochgeschwindigkeits-Drahtlosmodus arbeitet.

[0070] Der Beispiel-Sensorknotenaktualisierungsmanager **136** weist einen Beispiel-Aktualisierungsübermittler **326** auf. Im dargestellten Beispiel stellt der Aktualisierungsübermittler **326** Mittel zum Übermitteln der Firmwareaktualisierung(en) **202** zum/zu den zu aktualisierenden Sensorknoten bereit. Der Aktualisierungsübermittler **326** greift auf die Firmwareaktualisierung(en) **202**, welche in der Datenbank **300** von **Fig. 3** gespeichert ist/sind, zu. Der Aktualisierungsübermittler **326** greift auch auf die Firmwareanweisung(en) **211** zu. Der Aktualisierungsübermittler **326** weist den Kommunikator **308** an, die Firmwareaktualisierung(en) **202** basierend auf den Firmwareanweisungen **211** zum ersten Sensorknoten **104** zu übermitteln. Somit ermöglicht der Sensorknotenaktualisierungsmanager **136** von **Fig. 3** FOTA-Aktualisierungen am ersten Sensorknoten **104**.

[0071] Der Beispiel-Sensorknotenaktualisierungsmanager **136** von **Fig. 3** weist einen Beispiel-Aktualisierungsmonitor **328** auf. Im dargestellten Beispiel stellt der Aktualisierungsmonitor **328** Mittel zum Überwachen des Zustands der Installation der Firmwareaktualisierung(en) **202** am/an den Sensorknoten (z.B. dem ersten Sensorknoten **104** von **Fig. 1**) bereit. Zum Beispiel erzeugt der Aktualisierungsmonitor **328** Statusanfragen, welche an den Prozessor **112** des ersten Sensorknotens **104** übermittelt werden. Als Reaktion auf die Statusabfragen sendet der erste Sensorknoten **104** Mitteilungen bezüglich des Installationsfortschritts der Firmwareaktualisierung(en) **202**. Basierend auf dem Status der Firmwareaktualisierungsinstallation erzeugt der Aktualisierungsmonitor **328** die Statusmitteilung(en) **218**, welche durch den Kommunikator **308** von **Fig. 3** zum UAV-Manager **130** von **Fig. 2** übermittelt wird/werden, wie oben in Zusammenhang mit **Fig. 2** offenbart. Der Aktualisierungsmonitor **328** kann die Statusmitteilungen **218** gemäß einer durch Benutzereingaben, welche der UAV-Manager **130** und/oder der Sensorknotenaktua-

lisierungsmanager **136** empfängt, definierten Häufigkeit (z.B. regelmäßig, nur wenn ein Fehler auftritt) erzeugen.

[0072] Wenn die FOTA-Aktualisierung am ersten Sensorknoten **104** beendet ist, startet der erste Sensorknoten **104** wie oben offenbart neu, um die Installation der Firmwareaktualisierung abzuschließen und die Verwendung der aktualisierten Firmware einzuleiten. Ferner sendet der erste Sensorknoten **104** eine Mitteilung zum Beispiel-Aktualisierungsmonitor **328** von **Fig. 3**, welche bestätigt, dass die FOTA-Aktualisierung abgeschlossen ist. Der Beispiel-Aktualisierungsmonitor **328** bestätigt, dass die FOTA-Aktualisierung erfolgreich war (z.B. durch Überprüfen auf irgendwelche Fehlermeldungen und Überprüfen der Lösung dieser Fehler).

[0073] Wenn der Aktualisierungsmonitor **328** bestätigt, dass die FOTA-Aktualisierung erfolgreich war, erzeugt der Beispiel-Sensorknotenverbindungsmanager **322** eine Beispiel-RF-Schaltauslösermitteilung **324**, welche den ersten Sensorknoten **104** anweist, vom Hochgeschwindigkeits-Drahtlosmodus in den Energiespar-Drahtlosmodus umzuschalten. Ferner weist der Beispiel-Navigator **302** das UAV **128** an, zum nächsten Wegpunkt auf der Karte (z.B. der zweiten Straßenlampe **106**) zu fliegen, oder zur UAV-Basis zurückzukehren. Somit stellt der Beispiel-Sensorknotenaktualisierungsmanager **136** eine autonome Aktualisierung des/der IoT-Objektsensorknoten in einem IoT-Netzwerk über das UAV **128** bereit.

[0074] Wie oben erwähnt, können in manchen Beispielen eine oder mehrere Komponenten des UAV-Managers **130** durch den Sensorknotenaktualisierungsmanager **136** umgesetzt werden. Zum Beispiel kann der Kartengenerator **210** durch den Sensorknotenaktualisierungsmanager **136** von **Fig. 3** umgesetzt werden, um die Karte(n) **212** am UAV **128** zu erstellen. Ebenso können eine oder mehrere Komponenten des Sensorknotenaktualisierungsmanagers **136** durch den UAV-Manager **130** umgesetzt werden. Zum Beispiel kann der Kameramanager **304** durch den UAV-Manger **130** umgesetzt werden, um die Kamera(s) **142** zum Beispiel über die Benutzervorrichtung **124** zu steuern.

[0075] Während in **Fig. 3** ein Beispiel der Umsetzung des Beispiel-Sensorknotenaktualisierungsmanagers **136** dargestellt ist, können eines/einer/eine oder mehrere der Elemente, Prozesse und/oder Vorrichtungen, welche in **Fig. 3** dargestellt sind, kombiniert, geteilt, umgestellt, ausgelassen, weggelassen und/oder auf irgendeine andere Weise umgesetzt werden. Ferner können die Beispiel-Datenbank **300**, der Beispiel-Navigator **302**, der Beispiel-Kameramanager **304**, der Beispiel-Kommunikator **308**, der Beispiel-IoT-Objektidentifikator **314**, der Beispiel-Sensorknotenverbindungsmanager **322**, der

Beispiel-Aktualisierungsübermittler **326**, der Beispiel-Aktualisierungsmonitor **328** und/oder allgemeiner der Beispiel-Sensorknoten-Aktualisierungsmanager **136** von **Fig. 3** durch Hardware, Software, Firmware und/oder irgendeine Kombination von Hardware, Software und/oder Firmware umgesetzt werden. Somit könnte zum Beispiel die Beispiel-Datenbank **300**, der Beispiel-Navigator **302**, der Beispiel-Kameramanager **304**, der Beispiel-Kommunikator **308**, der Beispiel-IoT-Objektidentifikator **314**, der Beispiel-Sensorknotenverbindungsmanager **322**, der Beispiel-Aktualisierungsübermittler **326**, der Beispiel-Aktualisierungsmonitor **328** und/oder allgemeiner der Beispiel-Sensorknotenaktualisierungsmanager **136** von **Fig. 3** durch eine oder mehrere analoge oder digitale Schaltungen, Logikschaltungen, programmierbare Prozessoren, anwendungsspezifische integrierte Schaltungen (ASIC(s)), programmierbare Logikvorrichtung(en) (PLD(s)) und/oder feldprogrammierbare Logikvorrichtung(en) (FPLD(s)) umgesetzt werden. Beim Lesen irgendeines der Vorrichtungs- oder Systemansprüche dieses Patents zur Abdeckung einer reinen Software- und/oder Firmwareumsetzung ist hiermit mindestens das Beispiel, die Beispiel-Datenbank **300**, der Beispiel-Navigator **302**, der Beispiel-Kameramanager **304**, der Beispiel-Kommunikator **308**, der Beispiel-IoT-Objektidentifikator **314**, der Beispiel-Sensorknotenverbindungsmanager **322**, der Beispiel-Aktualisierungsübermittler **326**, der Beispiel-Aktualisierungsmonitor **328** und/oder allgemeiner der Beispiel-Sensorknotenaktualisierungsmanager **136** von **Fig. 3** ausdrücklich derart definiert, dass er eine nichtflüchtige computerlesbare Speichervorrichtung oder Speicherplatte, wie zum Beispiel einen Speicher, eine DVD (digitale versatile Disk), eine CD (Kompaktdiskette), eine Blu-ray-Disk, etc., welche die Software und/oder Firmware aufweist, aufweist. Ferner kann der Beispiel-Sensorknotenaktualisierungsmanager **136** von **Fig. 1** und **Fig. 3** auch ein/eine/einen oder mehrere Elemente, Prozesse und/oder Vorrichtungen zusätzlich zu, oder anstelle von, jenen, die in **Fig. 1** und **Fig. 3** dargestellt sind, aufweisen, und/oder kann mehr als eines/einen/eine der oder sämtliche der dargestellten Elemente, Prozesse und Vorrichtungen aufweisen.

[0076] Ablaufdiagramme, welche maschinenlesbare Beispielanweisungen zum Umsetzen des Beispielsystems **100** und/oder Komponenten davon, welche in **Fig. 1 - Fig. 3** dargestellt sind, darstellen, sind in **Fig. 4 - Fig. 6** gezeigt. In diesen Beispielen umfassen die maschinenlesbaren Anweisungen ein oder mehrere Programme zur Ausführung durch einen oder mehrere Prozessoren, wie zum Beispiel die Prozessoren **130**, **136**, welche in den Beispiel-Prozessorplattformen **700**, **800**, die unten in Verbindung mit **Fig. 7** und **Fig. 8** erörtert werden, gezeigt sind. Das/ die Programm(e) kann/können in einer Software eingebettet sein, welche auf einem nichtflüchtigen computerlesbaren Speichermedium, wie zum Beispiel ei-

ner CD-ROM, einer Diskette, einer Festplatte, einer DVD (digitale versatile Disk) einer Blu-ray-Disk oder einem dem/den Prozessor(en) **130**, **136** zugewiesenen Speicher, gespeichert ist, jedoch kann das gesamte Programm/können die gesamten Programme und/oder Teile davon alternativ dazu auch durch eine andere Vorrichtung als den/die Prozessor(en) **130**, **136** ausgeführt und als Firmware oder dedizierte Hardware umgesetzt sein. Ferner sind die/ist das Beispiel-Programm(e) zwar unter Bezugnahme auf die in **Fig. 4 - Fig. 6** dargestellten Ablaufdiagramme beschrieben, alternativ dazu können jedoch viel andere Verfahren zum Umsetzen des Beispielsystems **100** und/oder Komponenten desselben, welche in **Fig. 1 - Fig. 3** dargestellt sind, verwendet werden. Zum Beispiel kann die Reihenfolge der Ausführung der Blöcke verändert werden und/oder einige der beschriebenen Blöcke können geändert, weggelassen oder kombiniert werden. Darüber hinaus oder alternativ dazu kann irgendeiner oder sämtliche der Blöcke durch eine oder mehrere Hardwareschaltungen (z.B. diskrete und/oder integrierte analoge und/oder digitale Schaltungen, eine feldprogrammierbare Gattermatrix (FPGA), eine anwendungsspezifische integrierte Schaltung (ASIC), einen Komparator, einen Operationsverstärker (Op-Amp), eine Logikschaltung, etc.), welche dafür strukturiert sind, die entsprechende Operation ohne Ausführung von Software oder Firmware auszuführen, umgesetzt werden.

[0077] Wie oben erwähnt können die Beispiel-Prozesse von **Fig. 4 - Fig. 6** umgesetzt werden, indem kodierte Anweisungen verwendet werden (z.B. computer- und/oder maschinenlesbare Anweisungen), welche auf einem nichtflüchtigen computer- und/oder maschinenlesbaren Medium, wie zum Beispiel einem Festplattenlaufwerk, einem Flashspeicher, einem Festwertspeicher, einer Kompaktdisk, einer digitalen versatilen Disk, einem Cache, einem Direktzugriffsspeicher und/oder irgendeiner anderen Speichervorrichtung oder Speicherplatte, in welcher Informationen für eine beliebige Dauer (z.B. für längere Zeiträume, permanent, für kurze Zeiträume, zum vorübergehenden Zwischenspeichern und/oder zum Caching der Informationen) gespeichert werden, gespeichert sind. Wie hierin verwendet ist der Begriff nichtflüchtiges computerlesbares Medium ausdrücklich derart definiert, dass er jegliche Art von computerlesbarer Speichervorrichtung und/oder Speicherplatte einschließt und sich ausbreitende Signale und Übertragungsmedien ausschließt. „Einschließen“ (aufweisen) und „ausschließen“ (und sämtliche Formen und Zeiten davon) sind hierin als offene Begriffe verwendet. Immer wenn ein Anspruch irgendetwas nach irgendeiner Form von „einschließen/aufweisen“ oder „umfassen“ (z.B. umfasst, weist auf, umfassend, aufweisend, etc.) auflistet, versteht sich, dass zusätzliche Elemente, Begriffe etc. vorhanden sein können, ohne außerhalb des Umfangs des entsprechenden Anspruchs zu fallen. Wie hierin ver-

wendet, ist die Verwendung der Formulierung „mindestens“ als Übergangsbegriff in einem Oberbegriff eines Anspruchs ebenso offen bzw. unbeschränkt, wie die Begriffe „umfassend“ und „aufweisend“.

[0078] Fig. 4 ist ein Ablaufdiagramm maschinenlesbarer Beispielanweisungen, welche bei ihrer Ausführung bewirken, dass der Beispiel-Sensorknotenaktualisierungsmanager 136 von Fig. 1 und/oder 3 einem Sensorknoten (z.B. dem ersten Sensorknoten 104 von Fig. 1), welcher einem IoT-Objekt oder einer Vorrichtung (z.B. der ersten Straßenlampe 102 von Fig. 1) in einem IoT-Netzwerk (z.B. dem IoT-Netzwerk 126 von Fig. 1) zugeordnet ist, eine Drahtlos-Aktualisierung (z.B. eine FOTA-Aktualisierung) bereitstellt. Im Beispiel von Fig. 4 kann die FOTA-Aktualisierung dem Sensorknoten über ein UAV (z.B. das UAV 128 von Fig. 1) übermittelt werden. Die Beispielanweisungen von Fig. 4 können zum Beispiel durch den Prozessor 134 von Fig. 1 ausgeführt werden, um den Sensorknotenaktualisierungsmanager 136 von Fig. 1 und/oder 3 umzusetzen.

[0079] Der Beispiel-Aktualisierungsübermittler 326 des Sensorknotenaktualisierungsmanagers 136 von Fig. 3 greift auf die Firmwareaktualisierung(en) 202, welche einem oder mehreren Sensorknoten in einem IoT-Netzwerk zu übermitteln ist/sind, sowie die Firmwareanweisung(en) 211 zu (Block 400). In manchen Beispielen wird/werden die Firmwareaktualisierung(en) 202 durch den Firmwareaktualisierungsmanager 206 des UAV-Managers 130 von Fig. 2 erzeugt und zum Sensorknotenaktualisierungsmanager 136 von Fig. 3 übertragen. Die Firmwareaktualisierung(en) 202 und die Firmwareaktualisierungsanweisungen 211 werden in der Datenbank 300 von Fig. 3 gespeichert.

[0080] Der Beispiel-Navigator 302 erzeugt Anweisungen zum Navigieren des UAV 128 zum/zu den IoT-Objekt(en), welche(s) den/die zu aktualisierenden Sensorknoten aufweist/aufweisen, und der IoT-Objektidentifikator 314 identifiziert das/die IoT-Objekt(e) (Block 402). Der Beispiel-Navigator 302 erzeugt die Anweisungen zum Navigieren des UAV 128 basierend auf Karte(n) 212, welche (eine) Route(n) für den Flug des UAV 128 zum Erreichen des/der IoT-Objekt(e) aufweist/aufweisen (Block 402). Die Karte(n) 212 können durch den Beispiel-Kartengenerator 210 des UAV-Managers 130 von Fig. 2 erzeugt und zum Sensorknotenmanager 136 von Fig. 3 übertragen werden. In anderen Beispielen wird der Kartengenerator 210 durch den Sensorknotenmanager 136 von Fig. 3 umgesetzt.

[0081] Während des Flugs erzeugt/erzeugen die Beispielkamera(s) 142 des UAV 128 Bilddaten 310, welche Darstellungen der IoT-Objekte im IoT-Netzwerk 126 aufweisen. Der Beispiel-Kameramanager 304 verarbeitet die Bilddaten 310 und die Tiefenda-

ten 312, welche durch die Kamera(s) 142 erstellt worden sind, um ausgerichtete Bilddaten 313 zu erzeugen. Das/die IoT-Objekt(e) im Beispiel-IoT-Netzwerk 126 (z.B. die Straßenlampen 102, 106) weist/weisen jeweils Kennungen 105, 107, wie zum Beispiel graphische Kennungen, auf. Die Kennungen 105, 107 sind in den Bilddaten 310 und/oder den ausgerichteten Bilddaten 313 abgebildet. Der Beispiel-IoT-Objektidentifikator 314 identifiziert ein IoT-Objekt, welches einen zu aktualisierenden Sensorknoten aufweist, basierend auf IoT-Objektidentifikationsregeln 316 (z.B. die erste Straßenlampe 102, welche den ersten Sensorknoten 104 aufweist). Zum Beispiel vergleicht der IoT-Objektidentifikator 314 die Kennung(en), welche in den Bilddaten 310, 313 abgebildet ist/sind, mit bekannten Kennungen für IoT-Objekte, welche zu aktualisierende Sensorknoten aufweisen (z.B. die IoT-Objektidentifikationsdaten 209, welche in der Datenbank 300 von Fig. 3 gespeichert sind).

[0082] Wenn der IoT-Objektidentifikator 314 das IoT-Objekt, welches den zu aktualisierenden Sensorknoten aufweist, identifiziert hat, übermittelt der Aktualisierungsübermittler 326 die Firmwareaktualisierung 202 zum Sensorknoten (Block 404). In manchen Beispielen arbeitet der Sensorknoten (z.B. der Sensorknoten 104 von Fig. 1) in einem Energiespar-Drahtlosmodus (z.B. Kommunikation über das Sub-1-GHz-Frequenzband), um Sensordaten zu sammeln und zu übermitteln. Der Sensorknotenverbindungsmanager 322 sendet (eine) RF-Schaltauslösermitteilung(en) 324 über das erste (z.B. Niedriggeschwindigkeits-) RF-Modul des UAV 128 zum Sensorknoten. Die Mitteilung(en) 324 weist/weisen den Sensorknoten an, vom Betrieb im Energiespar-Drahtlosmodus auf einen Betrieb im Hochgeschwindigkeits-Drahtlosmodus (z.B. Kommunikation über das 2,4-GHz-Frequenzband) umzuschalten. Wenn der Sensorknoten auf den Betrieb im Hochgeschwindigkeits-Drahtlosmodus umgeschaltet hat, richtet der Aktualisierungsübermittler 326 über das erste (z.B. Hochgeschwindigkeits-) RF-Modul 138 des UAV 128 eine Kommunikation mit dem Sensorknoten ein. Der Aktualisierungsübermittler 326 übermittelt die Firmwareaktualisierung 202 über das erste RF-Modul 138 zum Sensorknoten 104 (z.B. dem Prozessor 112 des Sensorknotens 104).

[0083] Wenn die Firmwareaktualisierung 202 am Sensorknoten abgeschlossen ist, sendet der Sensorknotenverbindungsmanager 322 (eine) RF-Schaltauslösermitteilung(en) 324 über das erste (z.B. Hochgeschwindigkeits-) RF-Modul 138 an den Sensorknoten, um den Sensorknoten anzuweisen, zurück in den Energiespar-Drahtlosmodus zu schalten. Die Beispielanweisungen von Fig. 4 enden, wenn das UAV 128 die Firmwareaktualisierung(en) zum/zu den Sensorknoten des/der IoT-Objekt(e), welche(s) die Ak-

tualisierung(en) erhalten sollen, übermittelt hat (Block 406).

[0084] Fig. 5 ist ein Ablaufdiagramm zweiter maschinenlesbarer Beispielanweisungen, welche bei ihrer Ausführung bewirken, dass der Beispiel-Sensorknotenaktualisierungsmanager 136 von Fig. 1 und/oder 3 einem Sensorknoten (z.B. dem ersten Sensorknoten 104 von Fig. 1), welcher einem IoT-Objekt oder einer Vorrichtung (z.B. der ersten Straßenlampe 102 von Fig. 1) in einem IoT-Netzwerk (z.B. dem IoT-Netzwerk 126 von Fig. 1) zugeordnet ist, eine Drahtlos-Aktualisierung (z.B. eine FOTA-Aktualisierung) bereitstellt. Im Beispiel von Fig. 5 kann die FOTA-Aktualisierung dem Sensorknoten über ein UAV (z.B. das UAV 128 von Fig. 1) übermittelt werden. Die Beispielanweisungen von Fig. 5 können zum Beispiel durch den Prozessor 134 von Fig. 1 ausgeführt werden, um den Sensorknotenaktualisierungsmanager 136 von Fig. 1 und/oder 3 umzusetzen.

[0085] Der Beispiel-Aktualisierungsübermittler 326 des Sensorknotenaktualisierungsmanagers 136 von Fig. 3 greift auf die Firmwareaktualisierung(en) 202, welche einem oder mehreren Sensorknoten in einem IoT-Netzwerk zu übermitteln ist/sind, sowie auf die Firmwareanweisung(en) 211 zu (Block 500). In manchen Beispielen werden die Firmwareaktualisierung(en) 202 und die Anweisung(en) 211 durch den Firmwareaktualisierungsmanager 206 des UAV-Managers 130 von Fig. 2 erzeugt und zum Sensorknotenaktualisierungsmanager 136 von Fig. 3 übertragen. Die Firmwareaktualisierung(en) 202 und die Firmwareaktualisierungsanweisung(en) 211 werden in der Datenbank 300 von Fig. 3 gespeichert.

[0086] Der Beispielnavigator 302 greift auf die durch den Beispiel-Kartengenerator 210 von Fig. 2 erstellten IoT-Objektkarte(n) 212 zu (Block 501). Die IoT-Karte(n) 212 wird/werden in der Datenbank 300 von Fig. 3 gespeichert. Die IoT-Karte(n) 212 weist/weisen (eine) Route(n) zum/zu den IoT-Objekt(en), welche (s) den/die zu aktualisierenden Sensorknoten aufweist/aufweisen, auf. Zum Beispiel erstellt der Kartengenerator 210 von Fig. 2 eine Karte 212, welche eine Route aufweist, die das UAV 128 fliegen soll, um die erste Straßenlampe 102, welche den ersten Sensorknoten 104 aufweist, zu erreichen. In manchen Beispielen erstellt der Kartengenerator 210 basierend auf einer Ermittlung durch den Firmwareaktualisierungsmanager 206 von Fig. 2, dass zum Beispiel der erste Sensorknoten 104 zu aktualisieren ist, und einer oder mehreren Zuordnungsregeln, welche in der Datenbank 200 von Fig. 2 gespeichert sind, die Karte(n) 212 automatisch. In manchen Beispielen erstellt der Kartengenerator 210 die Karte(n) 212 basierend auf Benutzereingaben, welche zum Beispiel über die Benutzervorrichtung 124 empfangen werden. Die Benutzereingabe(n) kann/können zum Beispiel bevorzugte UAV-Routen, Zeitbeschränkun-

gen, etc. aufweisen. In manchen Beispielen wird der Kartengenerator 210 durch den UAV-Manager 130 von Fig. 2 umgesetzt und die Karte(n) 212 wird/werden zum Sensorknotenaktualisierungsmanager 136 von Fig. 2 übertragen. In anderen Beispielen wird der Kartengenerator 210 durch den Sensorknotenaktualisierungsmanager 136 von Fig. 3 umgesetzt.

[0087] Der Navigator 302 des Sensorknotenaktualisierungsmanagers 136 erzeugt Anweisungen zum Navigieren des UAV 128 zum/zu den IoT-Objekt(en), welche(s) den/die zu aktualisierenden Sensorknoten aufweist/aufweisen, basierend auf der/den Karte(n) 212 (Block 502). In manchen Beispielen kommuniziert der Navigator 302 mit Autopilotsoftware des UAV 128, um das UAV 128 basierend auf der/den Karte(n) 212 zu navigieren. In manchen Beispielen erzeugt der Navigator 302 Positionsdaten 303 in Bezug auf eine Position des UAV 128 relativ zu einem oder mehreren Wegpunkten in der Karte (212) (z.B. in Echtzeit).

[0088] Der Beispiel-Kameramanager 304 von Fig. 3 erzeugt Kameraauslöseranweisungen 306, welche die Kamera(s) 142 des UAV 128 anweisen, während des Flugs Bilddaten 310 und/oder Tiefendaten 312 (z.B. Messungen des Abstands des UAV 128 zum Boden) zu erzeugen (Block 504). Zum Beispiel weist der Kameramanager 304 die Kamera(s) 142 zum Beispiel basierend auf den UAV-Positionsdaten 303, welche angeben, dass sich das UAV 128 innerhalb eines Grenzabstands vom IoT-Objekt, welches den zu aktualisierenden Sensorknoten aufweist (z.B. der ersten Straßenlampe 102) befindet, an, die Bilddaten 310 und/oder die Tiefendaten 312 zu erzeugen. Der Kameramanager 304 verarbeitet die Bilddaten 310 und die Tiefendaten 312, um ausgerichtete Bilddaten 313 (z.B. 3D-Bilddaten) zu erzeugen.

[0089] Der Beispiel IoT-Objektidentifikator 314 identifiziert (ein) IoT-Objekt(e) von Interesse, oder (ein) IoT-Objekt(e), welche(s) ähnlich dem IoT-Objekt ist/sind, das den zu aktualisierenden Sensorblock aufweist (Block 506). In manchen Beispielen erkennt der IoT-Objektidentifikator 314 das/die IoT-Objekt(e) basierend auf den IoT-Objektidentifikationsregeln 316, den Bilddaten 310, den ausgerichteten Bilddaten 313 und den Referenzbilddaten 320. Zum Beispiel vergleicht der IoT-Objektidentifikator 314 die Bilddaten 310 und/oder die ausgerichteten Bilddaten 313 mit Referenzbilddaten 320, um die in den Bilddaten erfassten IoT-Objekte zu identifizieren. Basierend auf dem Vergleich erkennt der IoT-Objektidentifikator 314, dass das UAV 128 in der Nähe von IoT-Objekten angeordnet ist, welche den zu aktualisierenden Sensorknoten aufweisen könnten. Wenn der erste Sensorknoten 104 der ersten Straßenlampe 102 zu aktualisieren ist, erkennt der IoT-Objektidentifikator 314 zum Beispiel, dass sich das UAV 128 in der Nähe ei-

ner oder mehrere Straßenlampe(n), anstatt zum Beispiel eines Stoppschildes, befindet.

[0090] Der IoT-Objektidentifikator **314** analysiert die Bilddaten **310**, **313**, um das IoT-Objekt, welches den zu aktualisierenden Sensorknoten aufweist, zu identifizieren (Block **508**). Zum Beispiel ermittelt der IoT-Objektidentifikator **314**, ob irgendwelche der Kennungen des/der in den Bilddaten **310**, **313** erfassten IoT-Objekt(e) der Kennung **105**, welche basierend auf den IoT-Objektidentifikationsdaten **209** der ersten Straßenlampe **102** zugeordnet sind, entsprechen. In manchen Beispielen identifiziert der IoT-Objektidentifikator **314** basierend auf den Positionsdaten **303** und/oder der/den Karte(n) **212** das IoT-Objekt, welches den zu aktualisierenden Sensorknoten aufweist. In anderen Beispielen identifiziert der IoT-Objektidentifikator **314** basierend auf einer Stärke eines vom/von den Sensorknoten ausgesendeten Signals das IoT-Objekt, welches den/die zu aktualisierenden Sensorknoten aufweist.

[0091] Der IoT-Objektidentifikator **314** analysiert weiterhin die Bilddaten **310**, **313**, bis der IoT-Objektidentifikator **314** das IoT-Objekt, welches den zu aktualisierenden Sensorknoten aufweist (z.B. die erste Straßenlampe **102** von **Fig. 1**), erkennt. Der Beispielnavigator **302** von **Fig. 3** positioniert das UAV **128** in der Nähe des IoT-Objekts (Block **510**). Zum Beispiel verwendet der Navigator **302** die ausgerichteten Bilddaten **313** und/oder die Positionsdaten **303**, um das UAV **128** derart zu positionieren, dass es in der Nähe der ersten Straßenlampe **102** schwebt.

[0092] Der Beispiel-Sensorknotenverbindungsmanager **322** weist den Sensorknoten an, vom Betrieb in einem ersten oder Energiespar-Drahtlosmodus (z.B. Kommunikation über das Sub-1-GHz-Frequenzband) in einen zweiten oder Hochgeschwindigkeits-Drahtlosmodus (z.B. Kommunikation über das 2,4-GHz-Frequenzband, BLE) umzuschalten (Block **512**). Zum Beispiel erzeugt der Sensorknotenverbindungsmanager **322** eine RF-Schaltauslösermitteilung **324**, welche durch den Kommunikator **308** von **Fig. 3** zum ersten Sensorknoten **104** übermittelt wird, wenn der Sensorknoten im Energiespar-Drahtlosmodus arbeitet (z.B. wird die Mitteilung über das Sub-1-GHz-Frequenzband zum ersten Sensorknoten übermittelt). Als Reaktion auf die RF-Schaltauslösermitteilung **324** schaltet das Dualband-RF-Modul **114** des ersten Sensorknotens **104** vom Energiespar-Drahtlosmodus in den Hochgeschwindigkeits-Drahtlosmodus um.

[0093] Wenn der Sensorknoten im zweiten oder Hochgeschwindigkeits-Drahtlosmodus arbeitet, richtet der Sensorknotenverbindungsmanager **322** eine Kommunikation (z.B. eine Punkt-zu-Punkt-Verbindung über BLE) mit dem Sensorknoten ein (Block **514**). Der Aktualisierungsübermittler **326** übermittelt

die Firmwareaktualisierung(en) **202** über eine Drahtlos-Aktualisierung zum Sensorknoten (Block **516**). In manchen Beispielen überwacht der Aktualisierungsmonitor **328** den Installationsfortschritt der Firmwareaktualisierung(en) und erzeugt (eine) Statusmitteilung(en) **218**, welche zum UAV-Statusauswerter **220** des UAV-Managers **130** von **Fig. 2** übertragen wird/werden. In manchen Beispielen werden die Firmwareaktualisierungen basierend auf der/den Statusmitteilungen **218**, welche angeben, dass eine jeweilige Paketaktualisierung abgeschlossen ist, in zwei oder mehreren Paketen übermittelt.

[0094] Wenn die Firmwareaktualisierung abgeschlossen ist (Block **518**), weist der Sensorknotenverbindungsmanager **322** den Sensorknoten an, vom zweiten oder Hochgeschwindigkeits-Drahtlosmodus in den ersten oder Energiespar-Drahtlosmodus umzuschalten (Block **520**). Zum Beispiel erzeugt der Sensorknotenverbindungsmanager **322** eine RF-Schaltauslösermitteilung **324**, welche über den Kommunikator **308** von **Fig. 3** zum ersten Sensorknoten **104** übertragen wird. Als Reaktion kehrt das RF-Modul **114** des ersten Sensorknotens **104** zum Betrieb im Energiespar-Drahtlosmodus zurück, um Sensordaten über den Sensor **110** zu erzeugen, und um die Sensordaten zum Beispiel zum Master-Knoten **116** im lokalen Sensornetzwerk **118** von **Fig. 1** zu übertragen.

[0095] Der Aktualisierungsübermittler **326** ermittelt, ob ein weiterer Sensorknoten zu aktualisieren ist (Block **522**). Falls ein weiterer Sensorknoten zu aktualisieren ist (z.B. der zweite Sensorknoten **108** der zweiten Straßenlampe **106** von **Fig. 1**), navigiert der Navigator **302** das UAV **128** zu dem IoT-Objekt, welches den zu aktualisierenden Sensorknoten aufweist. Wenn keine weiteren zu aktualisierenden Sensorknoten vorliegen, enden die Anweisungen von **Fig. 4** und das UAV **128** kann zu seiner Basis zurückkehren (Block **524**).

[0096] **Fig. 6** ist ein Ablaufdiagramm maschinenlesbarer Beispielanweisungen, welche bei ihrer Ausführung bewirken, dass der Beispiel-UAV-Manager **130** von **Fig. 1** und/oder 2 ein UAV (z.B. das UAV **128** von **Fig. 1**) anweist, einem Sensorknoten (z.B. dem ersten Sensorknoten **104** von **Fig. 1**), welcher einem IoT-Objekt oder einer Vorrichtung (z.B. der ersten Straßenlampe **102** von **Fig. 1**) in einem IoT-Netzwerk (z.B. dem IoT-Netzwerk **126** von **Fig. 1**) zugeordnet ist, eine Drahtlos-Aktualisierung (z.B. eine FOTA-Aktualisierung) zu übermitteln. Die Beispielanweisungen von **Fig. 6** können zum Beispiel durch den Prozessor **132** der Benutzervorrichtung **124** von **Fig. 1**, des Servers **122** von **Fig. 1** und/oder einer oder mehreren cloudbasierten Vorrichtungen **120** von **Fig. 1** ausgeführt werden, um den UAV-Manager **130** von **Fig. 1** und/oder 2 umzusetzen.

[0097] Der Beispiel-Firmwareaktualisierungsmanager **206** erzeugt eine oder mehrere Firmwareaktualisierungsanweisungen **211**, um das Beispiel-UAV **128** anzuweisen, zu einem oder mehreren Sensorknoten zu fliegen, um dem/den Sensorknoten eine oder mehrere Firmwareaktualisierungen zu übermitteln (Block **600**). Die Firmwareaktualisierungsanweisung(en) **211** können auf Benutzereingaben zum Beispiel in Bezug auf den/die zu aktualisierenden Sensorknoten, eine Zeitlinie für das Übermitteln der Aktualisierung(en), wie die Aktualisierung(en) zu übermitteln sind (z.B. in einem oder mehreren Paketen), etc. basieren. In manchen Beispielen bestimmt der Firmwareaktualisierungsmanager **206** automatisch, welche/r Sensorknoten basierend auf Firmwareaktualisierungsregeln **208**, welche in der Datenbank **200** von **Fig. 2** gespeichert sind, zu aktualisieren ist/sind. Der Firmwareaktualisierungsmanager **206** erzeugt IoT-Objektidentifikationsdaten **209**, welche die Kennung(en) **105**, **107** des/der IoT-Objekt(e), welche die zu aktualisierenden Sensorknoten aufweisen, aufweisen.

[0098] Der Beispiel-Kartengenerator **210** von **Fig. 2** erzeugt eine oder mehrere Karten **212** für das/die IoT-Objekt(e), welche(s) den/die zu aktualisierenden Sensorknoten aufweist/aufweisen (z.B. die erste Straßenlampe **102**, welche den ersten Sensorknoten **104** aufweist) (Block **402**). Der Kartengenerator **210** erzeugt die Karte(n) **212** basierend auf Zuordnungsregeln **214** und IoT-Objektpositionsdaten **216**, welche in der Datenbank **200** von **Fig. 2** gespeichert sind. Die Zuordnungsregel(n) **214** weisen zum Beispiel bevorzugte Routen für den Flug des UAV **128** auf. Die IoT-Objektpositionsdaten **216** weisen Position(en) des/der IoT-Objekts/Objekte, welche(s) den/die zu aktualisierenden Sensorknoten aufweist/aufweisen, auf. Die durch den Kartengenerator **210** erstellte(n) Karte(n) **212** weisen (eine) Route(n) auf, welche das UAV **128** fliegen soll, um das/die IoT-Objekt(e), welche(s) die zu aktualisierenden Sensorknoten aufweist/aufweisen, zu erreichen.

[0099] Der Beispielkommunikator **204** von **Fig. 2** übermitteln die Firmwareaktualisierungsanweisung(en) **211** (welche z.B. die IoT-Kennungsdaten **209** aufweisen) und die Karte(n) **212** zum UAV **128** zur Speicherung und/oder Ausführung durch den Beispiel-Sensorknotenaktualisierungsmanager **136** von **Fig. 3** (Block **604**). In manchen Beispielen übermitteln der Kommunikator **204** die Firmwareaktualisierung(en) **202** zum UAV **128**. In anderen Beispielen wird/werden die Firmwareaktualisierung(en) **202** über einen weiteren Prozessor zum Speichern auf das UAV **128** übertragen.

[0100] Während der Übermittlung der Firmwareaktualisierung(en) **202** zum/zu den Sensorknoten und/oder nachdem die Firmwareaktualisierung(en) abgeschlossen ist/sind, erzeugt der Beispiel-Aktualisie-

rungsmonitor **328** des Sensorknotenaktualisierungsmanagers **136** von **Fig. 2** Statusmitteilungen **218** und übermitteln die Statusmitteilungen vom UAV **128** zum UAV-Manager **130**. Der UAV-Statusauswerter **220** des UAV-Managers **130** von **Fig. 3** analysiert die Statusmitteilungen **218** (Block **606**). Basierend auf den Statusmitteilungen **218** ermittelt der UAV-Statusauswerter **220** zum Beispiel, ob die Firmwareaktualisierung(en) erfolgreich installiert worden ist/sind, ob während der Installation ein Fehler aufgetreten ist, etc.

[0101] Der Beispiel-UAV-Statusauswerter **220** ermittelt, ob basierend auf den Statusmitteilungen **218** Anweisungen erzeugt werden müssen (Block **608**) und erzeugt die Anweisungen (Block **610**), falls dies der Fall ist. Der UAV-Statusauswerter **220** kann zum Beispiel Anweisungen für das UAV **128** erzeugen, zur UAV-Basis zurückzukehren, falls während der Installation der Firmwareaktualisierung(en) ein Fehler aufgetreten ist. Als ein weiteres Beispiel kann der UAV-Statusauswerter **220** Anweisungen für das UAV **128** erzeugen, ein zweites Firmwareaktualisierungspaket zum/zu den Sensorknoten zu übermitteln, nachdem er basierend auf den Statusmitteilungen **218** festgestellt hat, dass das erste Firmwareaktualisierungspaket erfolgreich installiert worden ist.

[0102] Die Beispielanweisungen von **Fig. 6** enden, wenn keine weiteren Anweisungen vorliegen, welche der UAV-Manager **130** zum UAV **128** übermitteln soll (Block **612**).

[0103] **Fig. 7** ist ein Blockdiagramm einer Beispiel-Prozessorplattform **700**, welche in der Lage ist, eine oder mehrere der Anweisungen von **Fig. 4** und/oder 5 auszuführen, um den Sensorknotenaktualisierungsmanager **136** von **Fig. 1** und/oder 3 umzusetzen. Die Prozessorplattform **700** kann zum Beispiel ein Server, ein Personalcomputer, eine mobile Vorrichtung (z.B. ein Mobiltelefon, ein Smartphone, ein Tablet, wie zum Beispiel ein i-Pad™), ein persönlicher digitaler Assistent (PDA), eine Internetvorrichtung, eine Drohne oder irgendeine andere Art von Rechenvorrichtung sein.

[0104] Die Prozessorplattform **700** des dargestellten Beispiels weist einen Prozessor **136** auf. Der Prozessor **136** des dargestellten Beispiels ist eine Hardware. Zum Beispiel kann der Prozessor **136** durch eine oder mehrere integrierte Schaltungen, logische Schaltungen, Mikroprozessoren oder Steuerungen irgendeiner gewünschten Produktfamilie oder irgendeines gewünschten Herstellers umgesetzt sein. Der Hardwareprozessor kann eine halbleiterbasierte Vorrichtung (z.B. auf Siliziumbasis) sein. In diesem Beispiel setzt der Prozessor **136** den Beispielnavigator **302**, den Beispiel-Kameramanager **304**, den Beispiel-IoT-Objektidentifikator **314**, den Beispiel-Sensorknotenverbindungsmanager **322**, den Beispiel-Aktualisie-

rungsübermittler **326** und/oder den Beispiel-Aktualisierungsmonitor **328** des Sensorknotenaktualisierungsmanagers **136** um.

[0105] Der Prozessor **136** des dargestellten Beispiels weist einen lokalen Speicher **713** auf (z.B. einen Cache). Der Prozessor **136** des dargestellten Beispiels steht über einen Bus **718** in Kommunikation mit einem Hauptspeicher, welcher einen flüchtigen Speicher **714** und einen nichtflüchtigen Speicher **716** aufweist. Der flüchtige Speicher **714** kann durch synchronen dynamischen Direktzugriffsspeicher (SDRAM), dynamischen Direktzugriffsspeicher (DRAM), dynamischen RAMBUS-Direktzugriffsspeicher (RDRAM) und/oder irgendeine andere Art von Direktzugriffsspeichervorrichtung umgesetzt sein. Der nichtflüchtige Speicher **716** kann durch Flashspeicher und/oder irgendeine andere gewünschte Art von Speichervorrichtung umgesetzt sein. Der Zugriff auf den Hauptspeicher **714**, **716** wird durch eine Speichersteuerung gesteuert. Die Datenbank **300** des Sensorknotenaktualisierungsmanagers kann durch den Hauptspeicher **714**, **716** und/oder den lokalen Speicher **713** umgesetzt sein.

[0106] Die Prozessorplattform **700** des dargestellten Beispiels weist auch eine Schnittstellenschaltung **720** auf. Die Schnittstellenschaltung **720** kann durch irgendeine Art von Schnittstellenstandard, wie zum Beispiel eine Ethernet-Schnittstelle, einen universellen seriellen Bus (USB) und/oder eine PCI-Express-schnittstelle umgesetzt sein.

[0107] Im dargestellten Beispiel sind eine oder mehrere Eingabevorrichtungen **722** mit der Schnittstellenschaltung **720** verbunden. Die Eingabevorrichtung(en) **722** ermöglicht/ermöglichen einem Benutzer, Daten und/oder Befehle in den Prozessor **136** einzugeben. Die Eingabevorrichtung(en) kann/können zum Beispiel durch einen Schallsensor, ein Mikrofon, eine Kamera (Foto oder Video), eine Tastatur, einen Knopf, eine Maus, einen Berührungsbildschirm, ein Touchpad, eine Steuerkugel, Isopoint und/oder ein Spracherkennungssystem umgesetzt sein.

[0108] Eine oder mehrere Ausgabevorrichtungen **724** sind ebenfalls mit der Schnittstellenschaltung **720** des dargestellten Beispiels verbunden. Die Ausgabevorrichtungen **724** können zum Beispiel durch Anzeigevorrichtungen (z.B. eine lichtemittierende Diode (LED), eine organische lichtemittierende Diode (OLED), eine Flüssigkristallanzeige, eine Kathodenstrahlröhrenanzeige (CRT), einen Berührungsbildschirm, eine haptische Ausgabevorrichtung, einen Drucker und/oder Lautsprecher) umgesetzt sein. Die Schnittstellenschaltung **720** des dargestellten Beispiels weist somit typischerweise eine Graphiktreiberkarte, einen Graphiktreiberchip und/oder einen Graphiktreiberprozessor auf.

[0109] Die Schnittstellenschaltung **720** des dargestellten Beispiels weist auch eine Kommunikationsvorrichtung, wie zum Beispiel einen Sender, einen Empfänger, einen Senderempfänger, ein Modem und/oder eine Netzwerkschnittstellenkarte, auf, um den Datenaustausch mit externen Maschinen (z.B. Datenverarbeitungsvorrichtungen jeglicher Art) über ein Netzwerk **726** (z.B. eine Ethernet-Verbindung, eine digitale Teilnehmerleitung (DSL), eine Telefonleitung, Koaxialkabel, ein Mobiltelefonsystem, etc.) zu erleichtern. In diesem Beispiel ist der Kommunikator **308** durch die Schnittstellenschaltung **720** umgesetzt.

[0110] Die Prozessorplattform **700** des dargestellten Beispiels weist auch eine oder mehrere Massenspeichervorrichtungen **728** zum Speichern von Software und/oder Daten auf. Beispiele derartiger Massenspeichervorrichtungen **728** umfassen Diskettenlaufwerke, Festplatten, Kompaktdisklaufwerke, Blu-Ray-Disk-Laufwerke, RAID-Systeme und digitale versatile Disk- (DVD-) Laufwerke.

[0111] Die kodierten Anweisungen **732** von Fig. 4 und/oder 5 können in der Massenspeichervorrichtung **728**, im flüchtigen Speicher **714**, im nichtflüchtigen Speicher **716** und/oder auf einem herausnehmbaren, greifbaren computerlesbaren Speichermedium, wie zum Beispiel einer CD oder DVD, gespeichert sein.

[0112] Fig. 8 ist ein Blockdiagramm einer Beispiel-Prozessorplattform **800**, welche in der Lage ist, eine oder mehrere der Anweisungen von Fig. 6 auszuführen, um den UAV-Manager **130** von Fig. 1 und/oder 2 umzusetzen. Die Prozessorplattform **800** kann zum Beispiel ein Server, ein Personalcomputer, eine mobile Vorrichtung (z.B. ein Mobiltelefon, ein Smartphone, ein Tablet, wie zum Beispiel ein iPad™), ein persönlicher digitaler Assistent (PDA), eine Internetvorrichtung, eine Drohne oder irgendeine andere Art von Rechenvorrichtung sein.

[0113] Die Prozessorplattform **800** des dargestellten Beispiels weist einen Prozessor **130** auf. Der Prozessor **130** des dargestellten Beispiels ist eine Hardware. Zum Beispiel kann der Prozessor **130** durch eine oder mehrere integrierte Schaltungen, logische Schaltungen, Mikroprozessoren oder Steuerungen irgendeiner gewünschten Produktfamilie oder irgendeines gewünschten Herstellers umgesetzt sein. Der Hardwareprozessor kann eine halbleiterbasierte Vorrichtung (z.B. auf Siliziumbasis) sein. In diesem Beispiel setzt der Prozessor **130** den Beispiel-Firmwareaktualisierungsmanager **206**, den Beispiel-Kartengenerator **210** und/oder den Beispiel-UAV-Statusauswerter **220** des Beispiel-UAV-Managers **130** um.

[0114] Der Prozessor **130** des dargestellten Beispiels weist einen lokalen Speicher **813** auf (z.B. einen Cache). Der Prozessor **130** des dargestell-

ten Beispiels steht über einen Bus **818** in Kommunikation mit einem Hauptspeicher, welcher einen flüchtigen Speicher **814** und einen nichtflüchtigen Speicher **816** aufweist. Der flüchtige Speicher **814** kann durch synchronen dynamischen Direktzugriffsspeicher (SDRAM), dynamischen Direktzugriffsspeicher (DRAM), dynamischen RAMBUS-Direktzugriffsspeicher (RDRAM) und/oder irgendeine andere Art von Direktzugriffsspeichervorrichtung umgesetzt sein. Der nichtflüchtige Speicher **816** kann durch Flashspeicher und/oder irgendeine andere gewünschte Art von Speichervorrichtung umgesetzt sein. Der Zugriff auf den Hauptspeicher **814**, **816** wird durch eine Speichersteuerung gesteuert. Die Datenbank **200** des UAV-Managers kann durch den Hauptspeicher **814**, **816** und/oder den lokalen Speicher **813** umgesetzt sein.

[0115] Die Prozessorplattform **800** des dargestellten Beispiels weist auch eine Schnittstellenschaltung **820** auf. Die Schnittstellenschaltung **820** kann durch irgendeine Art von Schnittstellenstandard, wie zum Beispiel eine Ethernet-Schnittstelle, einen universellen seriellen Bus (USB) und/oder eine PCI-Express-schnittstelle umgesetzt sein.

[0116] Im dargestellten Beispiel sind eine oder mehrere Eingabevorrichtungen **822** mit der Schnittstellenschaltung **820** verbunden. Die Eingabevorrichtung(en) **822** ermöglicht/ermöglichen einem Benutzer, Daten und/oder Befehle in den Prozessor **130** einzugeben. Die Eingabevorrichtung(en) kann/können zum Beispiel durch einen Schallsensor, ein Mikrofon, eine Kamera (Foto oder Video), eine Tastatur, einen Knopf, eine Maus, einen Berührungsbildschirm, ein Touchpad, eine Steuerkugel, Isopoint und/oder ein Spracherkennungssystem umgesetzt sein.

[0117] Eine oder mehrere Ausgabevorrichtungen **824** sind ebenfalls mit der Schnittstellenschaltung **820** des dargestellten Beispiels verbunden. Die Ausgabevorrichtungen **824** können zum Beispiel durch Anzeigevorrichtungen (z.B. eine lichtemittierende Diode (LED), eine organische lichtemittierende Diode (OLED), eine Flüssigkristallanzeige, eine Kathodenstrahlröhrenanzeige (CRT), einen Berührungsbildschirm, eine haptische Ausgabevorrichtung, einen Drucker und/oder Lautsprecher) umgesetzt sein. Die Schnittstellenschaltung **820** des dargestellten Beispiels weist somit typischerweise eine Graphiktreiberkarte, einen Graphiktreiberchip und/oder einen Graphiktreiberprozessor auf.

[0118] Die Schnittstellenschaltung **820** des dargestellten Beispiels weist auch eine Kommunikationsvorrichtung, wie zum Beispiel einen Sender, einen Empfänger, einen Senderempfänger, ein Modem und/oder eine Netzwerkschnittstellenkarte, auf, um den Datenaustausch mit externen Maschinen (z.B.

Datenverarbeitungsvorrichtungen jeglicher Art) über ein Netzwerk **826** (z.B. eine Ethernet-Verbindung, eine digitale Teilnehmerleitung (DSL), eine Telefonleitung, Koaxialkabel, ein Mobiltelefonsystem, etc.) zu erleichtern. In diesem Beispiel ist der Kommunikator **204** durch die Schnittstellenschaltung **820** umgesetzt.

[0119] Die Prozessorplattform **800** des dargestellten Beispiels weist auch eine oder mehrere Massenspeichervorrichtungen **828** zum Speichern von Software und/oder Daten auf. Beispiele derartiger Massenspeichervorrichtungen **828** umfassen Diskettenlaufwerke, Festplatten, Kompaktdisklaufwerke, Blu-Ray-Disk-Laufwerke, RAID-Systeme und digitale versatile Disk- (DVD-) Laufwerke.

[0120] Die kodierten Anweisungen **832** von Fig. 6 können in der Massenspeichervorrichtung **828**, im flüchtigen Speicher **814**, im nichtflüchtigen Speicher **816** und/oder auf einem herausnehmbaren, greifbaren computerlesbaren Speichermedium, wie zum Beispiel einer CD oder DVD, gespeichert sein.

[0121] Aus den vorstehenden Ausführungen ergibt sich, dass die Beispielverfahren, Systeme und Vorrichtungen offenbart worden sind, um Drahtlos-Firmware- (FOTA-) Aktualisierungen unter Verwendung unbemannter Fluggeräte (UAV) zu Sensorknoten in einem IoT-Netzwerk zu übertragen. Offenbarte Beispiele speichern die am Sensorknoten zu installierende Firmwareaktualisierung auf dem UAV und weisen das UAV an, zu einem IoT-Objekt zu fliegen, welches den zu aktualisierenden Sensorknoten aufweist. Offenbarte Beispiel-UAVs kommunizieren automatisch mit dem Sensorknoten, um die Aktualisierung zu installieren. Offenbarte Beispiele bewirken, dass der Sensorknoten vom Betrieb in einem Energiespar-Drahtlosmodus zum Sammeln und Übermitteln von Sensordaten (z.B. Kommunikation über das Sub-1-GHz-Frequenzband) in einen Hochgeschwindigkeits-Drahtlosmodus (z.B. Kommunikation über das 2,4-GHz-Frequenzband) umschaltet. Offenbarte Beispiele übermitteln die FOTA-Aktualisierung über das UAV, wenn der Sensorknoten im Hochgeschwindigkeits-Drahtlosmodus arbeitet. Somit stellen offenbarte Beispiele eine effiziente Übermittlung der FOTA-Aktualisierung im Vergleich zu einer FOTA-Aktualisierung, welche übertragen wird, während der Sensorknoten im Energiespar-Drahtlosmodus arbeitet, bereit. Ferner bewältigt das Verwenden der hierin offenbarten Beispiel-UAV zum Übermitteln der FOTA-Aktualisierung wirksam die Anforderungen in Bezug auf Nähe zum Übertragen der FOTA-Aktualisierung über Drahtloskommunikationsprotokolle. Offenbarte Beispiele weisen den Sensorknoten an, zum Betrieb im Energiespar-Drahtlosmodus zurückzuschalten, wodurch sie ein autonomes Aktualisieren des Sensorknotens über das UAV bereitstellen.

[0122] Die folgende ist eine nicht ausschließliche Liste hierin offenbarer Beispiele. Im vorstehenden Text können auch andere Beispiele offenbart sein. Darüber hinaus können sämtliche der hierin offenbarten Beispiele als Ganzes oder in Teilen und/oder auf andere Weise modifiziert in Betracht gezogen werden.

[0123] Beispiel 1 weist ein unbemanntes Fluggerät aufweisend einen Aktualisierungsübermittler zum Zugreifen auf eine Firmwareaktualisierung, welche zu einem Sensorknoten in einem Netzwerk zu übermitteln ist, wobei der Sensorknoten mit einem Objekt verbunden ist; eine Kamera zum Erzeugen von Bilddaten; und einen Identifikator zum Identifizieren des Objekts basierend auf den Bilddaten, auf, wobei der Aktualisierungsübermittler dem Sensorknoten die Firmwareaktualisierung basierend auf der Identifizierung des Objekts übermittelt.

[0124] Beispiel 2 weist das unbemannte Fluggerät gemäß der Definition in Beispiel 1 auf, wobei der Identifikator das Objekt ferner basierend auf einer dem Objekt zugeordneten Kennung identifiziert, wobei der Identifikator die Kennung in den Bilddaten erfasst.

[0125] Beispiel 3 weist das unbemannte Fluggerät gemäß der Definition in Beispiel 1 oder Beispiel 2 auf, und weist ferner einen Sensorknotenverbindungsmanager auf, um den Sensorknoten anzuweisen, von einem ersten Drahtlosmodus in einen zweiten Drahtlosmodus umzuschalten, wobei dem zweiten Drahtlosmodus ein höheres Frequenzband zugewiesen ist, als dem ersten Drahtlosmodus.

[0126] Beispiel 4 weist das unbemannte Fluggerät gemäß der Definition in Beispiel 3 auf, wobei der Aktualisierungsübermittler die Firmwareaktualisierung übermittelt, wenn sich der Sensorknoten im zweiten Drahtlosmodus befindet.

[0127] Beispiel 5 weist das unbemannte Fluggerät gemäß der Definition in Beispiel 4 auf, wobei der Sensorknotenverbindungsmanager den Sensorknoten anweist, vom zweiten Drahtlosmodus in den ersten Drahtlosmodus umzuschalten, wenn die Firmwareaktualisierung abgeschlossen ist.

[0128] Beispiel 6 weist das unbemannte Fluggerät gemäß der Definition in Beispiel 3 auf, und weist ferner ein erstes Funkfrequenzmodul und ein zweites Funkfrequenzmodul auf, welches sich vom ersten Frequenzmodul unterscheidet, wobei das zweite Funkmodul in einem höheren Frequenzband überträgt, als das erste Funkfrequenzmodul, wobei der Sensorknotenverbindungsmanager eine Anweisung, welche den Sensorknoten anweist, in den zweiten Drahtlosmodus zu schalten, über das erste Funkfrequenzmodul sendet.

[0129] Beispiel 7 weist das unbemannte Fluggerät gemäß der Definition in Beispiel 6 auf, wobei der Aktualisierungsübermittler die Firmwareaktualisierung über das zweite Funkfrequenzmodul übermittelt.

[0130] Beispiel 8 weist das unbemannte Fluggerät gemäß der Definition in Beispiel 1 auf, und weist ferner einen Aktualisierungsmonitor zum Erzeugen einer Statusmitteilung für die Firmwareaktualisierung auf.

[0131] Beispiel 9 weist das unbemannte Fluggerät gemäß der Definition in Beispiel 8 auf, und weist ferner einen Kommunikator zum Übermitteln der Statusmitteilung an eine Benutzervorrichtung auf.

[0132] Beispiel 10 weist das unbemannte Fluggerät gemäß der Definition in Beispiel 1 auf, und weist ferner einen Kameramanager zum Aktivieren der Kamera zum Erzeugen der Bilddaten auf.

[0133] Beispiel 11 weist das unbemannte Fluggerät gemäß der Definition in den Beispielen 1, 2, 8 oder 10 auf, und weist ferner einen Navigator auf, um das unbemannte Fluggerät anzuweisen, basierend auf einer Karte zum Objekt zu fliegen.

[0134] Beispiel 12 weist das unbemannte Fluggerät von Beispiel 11 auf, wobei der Navigator Positionsdaten erzeugt und der Identifikator das Objekt ferner basierend auf den Positionsdaten identifiziert.

[0135] Beispiel 13 weist mindestens ein nichtflüchtiges computerlesbares Speichermedium auf, welches Anweisungen aufweist, die bei ihrer Ausführung bewirken, dass eine Maschine auf eine Firmwareaktualisierung zugreift, welche zu einem Sensorknoten in einem Netzwerk zu übermitteln ist, wobei der Sensorknoten mit einem Objekt verbunden ist; Bilddaten erzeugt; das Objekt basierend auf den Bilddaten identifiziert; und die Firmwareaktualisierung basierend auf der Identifizierung des Objekts zum Sensorknoten übermittelt.

[0136] Beispiel 14 weist das mindestens eine nichtflüchtige computerlesbare Speichermedium gemäß der Definition in Beispiel 13 auf, wobei das Objekt eine Kennung aufweist, und wobei die Anweisungen ferner bewirken, dass die Maschine die Kennung in den Bilddaten entdeckt und das Objekt basierend auf der Kennung identifiziert.

[0137] Beispiel 15 weist das mindestens eine nichtflüchtige computerlesbare Speichermedium gemäß der Definition in den Beispielen 13 oder 14 auf, wobei die Anweisungen ferner bewirken, dass die Maschine den Sensorknoten anweist, von einem ersten Drahtlosmodus in einen zweiten Drahtlosmodus umzuschalten, wobei dem zweiten Drahtlosmodus ein

höheres Frequenzband zugewiesen ist, als dem ersten Drahtlosmodus.

[0138] Beispiel 16 weist das mindestens eine nicht-flüchtige computerlesbare Speichermedium gemäß der Definition in Beispiel 15 auf, wobei die Anweisungen ferner bewirken, dass die Maschine die Firmwareaktualisierung übermittelt, wenn sich der Sensorknoten im zweiten Drahtlosmodus befindet.

[0139] Beispiel 17 weist das mindestens eine nicht-flüchtige computerlesbare Speichermedium gemäß der Definition in Beispiel 16 auf, wobei die Anweisungen ferner bewirken, dass die Maschine den Sensorknoten anweist, vom zweiten Drahtlosmodus in den ersten Drahtlosmodus umzuschalten, wenn die Firmwareaktualisierung abgeschlossen ist.

[0140] Beispiel 18 weist das mindestens eine nicht-flüchtige computerlesbare Speichermedium gemäß der Definition in Beispiel 15 auf, wobei die Anweisungen ferner bewirken, dass die Maschine eine Anweisung über ein erstes Funkfrequenzmodul sendet, um den Sensorknoten anzuweisen, in den zweiten Drahtlosmodus umzuschalten.

[0141] Beispiel 19 weist das mindestens eine nicht-flüchtige computerlesbare Speichermedium gemäß der Definition in Beispiel 18 auf, wobei die Anweisungen bewirken, dass die Maschine die Firmwareaktualisierung über ein zweites Funkfrequenzmodul übermittelt, wobei das zweite Funkfrequenzmodul in einem höheren Frequenzband überträgt als das erste Funkfrequenzmodul.

[0142] Beispiel 20 weist ein Verfahren auf, welches das Zugreifen auf eine Firmwareaktualisierung, welche zu einem Sensorknoten in einem Netzwerk zu übermitteln ist, wobei der Sensorknoten mit einem Objekt verbunden ist, das Erzeugen von Bilddaten, das Identifizieren des Objekts basierend auf den Bilddaten und das Übermitteln der Firmwareaktualisierung zum Sensorknoten basierend auf der Identifizierung des Objekts umfasst.

[0143] Beispiel 21 weist das Verfahren gemäß der Definition in Beispiel 20 auf, wobei das Objekt eine Kennung aufweist, und das Verfahren ferner das Erfassen der Kennung in den Bilddaten und das Identifizieren des Objekts basierend auf der Kennung umfasst.

[0144] Beispiel 22 weist das Verfahren gemäß der Definition in den Beispielen 20 oder 21 auf, und umfasst ferner das Anweisen des Sensorknotens, von einem ersten Drahtlosmodus in einen zweiten Drahtlosmodus umzuschalten, wobei dem zweiten Drahtlosmodus ein höheres Frequenzband zugewiesen ist als dem ersten Drahtlosmodus.

[0145] Beispiel 23 weist das Verfahren gemäß der Definition in Beispiel 22 auf, und umfasst ferner das Übermitteln der Firmwareaktualisierung, wenn sich der Sensorknoten im zweiten Drahtlosmodus befindet.

[0146] Beispiel 24 weist das Verfahren gemäß der Definition in Beispiel 23 auf, und umfasst ferner das Anweisen des Sensorknotens, vom zweiten Drahtlosmodus in den ersten Drahtlosmodus umzuschalten, wenn die Firmwareaktualisierung abgeschlossen ist.

[0147] Beispiel 25 weist das Verfahren gemäß der Definition in Beispiel 22 auf, und umfasst ferner das Senden einer Anweisung über ein erstes Funkfrequenzmodul, um den Sensorknoten anzuweisen, in den zweiten Drahtlosmodus umzuschalten.

[0148] Beispiel 26 weist das Verfahren gemäß der Definition in Beispiel 25 auf, und umfasst ferner das Übermitteln der Firmwareaktualisierung über ein zweites Funkfrequenzmodul, wobei das zweite Funkfrequenzmodul in einem höheren Frequenzband überträgt als das erste Funkfrequenzmodul.

[0149] Beispiel 27 weist ein System aufweisend einen Manager für ein unbemanntes Fluggerät zum Erzeugen einer Anweisung für ein unbemanntes Fluggerät, eine Firmwareaktualisierung zu einem Sensorknoten in einem Sensorknotennetzwerk zu übermitteln, und einen Sensorknotenaktualisierungsmanager auf. Der Sensorknotenaktualisierungsmanager navigiert als Reaktion auf die Anweisung vom Manager für das unbemannte Fluggerät das unbemannte Fluggerät zu einer Position in der Nähe des Sensorknotens, übermittelt eine Mitteilung an den Sensorknoten zum Umschalten von einem ersten Drahtlosmodus in einen zweiten Drahtlosmodus, und übermittelt die Firmwareaktualisierung, wenn sich der Sensorknoten im zweiten Drahtlosmodus befindet.

[0150] Beispiel 28 weist das System gemäß der Definition in Beispiel 27 auf, wobei der Manager für das unbemannte Fluggerät durch einen Prozessor einer Benutzervorrichtung umgesetzt wird.

[0151] Beispiel 29 weist das System gemäß der Definition in Beispiel 27 auf, wobei der Sensorknotenaktualisierungsmanager die Mitteilung zum Sensorknoten übermittelt, wenn der Sensorknoten im ersten Drahtlosmodus arbeitet.

[0152] Beispiel 30 weist das System gemäß der Definition in Beispiel 27 auf, wobei der Sensorknotenaktualisierungsmanager ein Objekt, welches den Sensorknoten aufweist, basierend auf Bilddaten und einer dem Objekt zugeordneten Kennung identifiziert.

[0153] Beispiel 31 weist das System gemäß der Definition in Beispiel 27 auf, wobei der Manager für

das unbemannte Fluggerät eine Karte erstellt, wobei der Sensorknotenaktualisierungsmanager das unbemannte Fluggerät basierend auf der Karte navigiert.

[0154] Beispiel 32 weist eine Vorrichtung aufweisend Mittel zum Navigieren eines unbemannten Fluggeräts zu einem Sensorknoten in einem Sensorknotennetzwerk, Mittel zum Anweisen des Sensorknotens, von einem ersten Drahtlosmodus in einen zweiten Drahtlosmodus umzuschalten, und Mittel zum Übermitteln einer Firmwareaktualisierung über das unbemannte Fluggerät zum Sensorknoten, wenn sich der Sensorknoten im zweiten Drahtlosmodus befindet, auf.

[0155] Beispiel 33 weist die Vorrichtung gemäß der Definition in Beispiel 32 auf, ferner aufweisend Mittel zum Identifizieren eines Objekts, welches den Sensorknoten aufweist, wobei die Mittel zum Identifizieren des Sensorknotens den Sensorknoten basierend auf der Identifizierung des Objekts identifizieren.

[0156] Beispiel 34 weist die Vorrichtung gemäß der Definition in Beispiel 33 auf, ferner aufweisend Mittel zum Erzeugen von Bilddaten, wobei die Mittel zum Identifizieren des Objektes das Objekt basierend auf den Bilddaten identifizieren.

[0157] Beispiel 35 weist die Vorrichtung gemäß der Definition in Beispiel 32 auf, wobei die Mittel zum Anweisen des Sensorknotens den Sensorknoten anweisen, vom zweiten Drahtlosmodus in den ersten Drahtlosmodus umzuschalten, nachdem die Mittel zum Übermitteln der Firmwareaktualisierung die Firmwareaktualisierung übermittelt haben.

[0158] Obwohl hierin gewisse Beispiel-Verfahren, Vorrichtungen und daraus hergestellte Artikel offenbart worden sind, ist der Umfang der Abdeckung durch dieses Patent nicht auf diese beschränkt. Vielmehr deckt dieses Patent sämtliche Verfahren, Vorrichtungen und daraus hergestellte Artikel ab, welche den Regeln entsprechend in den Umfang der Ansprüche dieses Patents fallen.

Patentansprüche

1. Unbemanntes Fluggerät, umfassend:
einen Aktualisierungsübermittler zum Zugreifen auf eine Firmwareaktualisierung, welche einem Sensorknoten in einem Netzwerk zu übermitteln ist, wobei der Sensorknoten mit einem Objekt verbunden ist, eine Kamera zum Erzeugen von Bilddaten; und
einen Identifikator, um das Objekt basierend auf den Bilddaten zu identifizieren, wobei der Aktualisierungsübermittler dem Sensorknoten die Firmwareaktualisierung basierend auf der Identifizierung des Objekts übermittelt.

2. Unbemanntes Fluggerät gemäß der Definition in Anspruch 1, wobei der Identifikator das Objekt ferner basierend auf einer dem Objekt zugeordneten Kennung identifiziert, wobei der Identifikator die Kennung in den Bilddaten erfasst.

3. Unbemanntes Fluggerät gemäß der Definition in den Ansprüchen 1 oder 2, ferner aufweisend einen Sensorknotenverbindungsmanager, um den Sensorknoten anzuweisen, von einem ersten Drahtlosmodus in einen zweiten Drahtlosmodus umzuschalten, wobei dem zweiten Drahtlosmodus ein höheres Frequenzband zugewiesen ist als dem ersten Drahtlosmodus.

4. Unbemanntes Fluggerät gemäß der Definition in Anspruch 3, wobei der Aktualisierungsübermittler die Firmwareaktualisierung übermittelt, wenn sich der Sensorknoten im zweiten Drahtlosmodus befindet.

5. Unbemanntes Fluggerät gemäß der Definition in Anspruch 4, wobei der Sensorknotenverbindungsmanager den Sensorknoten anweist, vom zweiten Drahtlosmodus in den ersten Drahtlosmodus umzuschalten, wenn die Firmwareaktualisierung abgeschlossen ist.

6. Unbemanntes Fluggerät gemäß der Definition in Anspruch 3, ferner aufweisend ein erstes Funkfrequenzmodul und ein zweites Funkfrequenzmodul, welches sich vom ersten Frequenzmodul unterscheidet, wobei das zweite Funkmodul in einem höheren Frequenzband überträgt, als das erste Funkfrequenzmodul, wobei der Sensorknotenverbindungsmanager eine Anweisung, welche den Sensorknoten anweist, in den zweiten Drahtlosmodus umzuschalten, über das erste Funkfrequenzmodul sendet.

7. Unbemanntes Fluggerät gemäß der Definition in Anspruch 6, wobei der Aktualisierungsübermittler die Firmwareaktualisierung über das zweite Funkfrequenzmodul übermittelt.

8. Unbemanntes Fluggerät gemäß der Definition in Anspruch 1, ferner aufweisend einen Aktualisierungsmonitor zum Erzeugen einer Statusmitteilung für die Firmwareaktualisierung.

9. Unbemanntes Fluggerät gemäß der Definition in Anspruch 8, ferner aufweisend einen Kommunikator zum Übermitteln der Statusmitteilung an eine Benutzervorrichtung.

10. Unbemanntes Fluggerät gemäß der Definition in Anspruch 1, ferner aufweisend einen Kameramanager, um die Kamera dazu zu aktivieren, die Bilddaten zu erzeugen.

11. Unbemanntes Fluggerät gemäß der Definition in einem der Ansprüche 1, 2, 8 oder 10, ferner auf-

weisend einen Navigator, um das unbemannte Fluggerät anzuweisen, basierend auf einer Karte zum Objekt zu fliegen.

12. Unbemanntes Fluggerät nach Anspruch 11, wobei der Navigator Positionsdaten erzeugt und der Identifikator das Objekt ferner basierend auf den Positionsdaten identifiziert.

13. Nichtflüchtiges computerlesbares Speichermedium bzw. nichtflüchtige computerlesbare Speichermedien aufweisend Anweisungen, welche bei ihrer Ausführung bewirken, dass eine Maschine:
auf eine Firmwareaktualisierung, welche einem Sensorknoten in einem Netzwerk zu übermitteln ist, zugreift, wobei der Sensorknoten mit einem Objekt verbunden ist;
 Bilddaten erzeugt;
 das Objekt basierend auf den Bilddaten identifiziert;
 und
 die Firmwareaktualisierung basierend auf der Identifikation des Objekts zum Sensorknoten übermittelt.

14. Nichtflüchtiges computerlesbares Speichermedium bzw. nichtflüchtige computerlesbare Speichermedien gemäß der Definition in Anspruch 13, wobei das Objekt eine Kennung aufweist, und wobei die Anweisungen ferner bewirken, dass die Maschine die Kennung in den Bilddaten entdeckt und das Objekt basierend auf der Kennung identifiziert.

15. Nichtflüchtiges computerlesbares Speichermedium bzw. nichtflüchtige computerlesbare Speichermedien gemäß der Definition in den Ansprüchen 13 oder 14, wobei die Anweisungen ferner bewirken, dass die Maschine den Sensorknoten anweist, von einem ersten Drahtlosmodus in einen zweiten Drahtlosmodus umzuschalten, wobei dem zweiten Drahtlosmodus ein höheres Frequenzband zugewiesen ist, als dem ersten Drahtlosmodus.

16. Nichtflüchtiges computerlesbares Speichermedium bzw. nichtflüchtige computerlesbare Speichermedien gemäß der Definition in Anspruch 15, wobei die Anweisungen ferner bewirken, dass die Maschine die Firmwareaktualisierung übermittelt, wenn sich der Sensorknoten im zweiten Drahtlosmodus befindet.

17. Nichtflüchtiges computerlesbares Speichermedium bzw. nichtflüchtige computerlesbare Speichermedien gemäß der Definition in Anspruch 16, wobei die Anweisungen ferner bewirken, dass die Maschine den Sensorknoten anweist, vom zweiten Drahtlosmodus in den ersten Drahtlosmodus umzuschalten, wenn die Firmwareaktualisierung abgeschlossen ist.

18. Verfahren, umfassend:
Zugreifen auf eine Firmwareaktualisierung, welche einem Sensorknoten in einem Netzwerk zu übermit-

telt ist, wobei der Sensorknoten mit einem Objekt verbunden ist,
Erzeugen von Bilddaten;
Identifizieren des Objekts basierend auf den Bilddaten; und
Übermitteln der Firmwareaktualisierung basierend auf der Identifikation des Objekts zum Sensorknoten.

19. System, umfassend:
einen Manager für ein unbemanntes Fluggerät zum Erzeugen einer Anweisung für ein unbemanntes Fluggerät, eine Firmwareaktualisierung zu einem Sensorknoten in einem Sensorknoten Netzwerk zu übermitteln; und
einen Sensorknotenaktualisierungsmanager, um als Reaktion auf die Anweisung des Managers für das unbemannte Fluggerät:
das unbemannte Fluggerät zu einer Position in der Nähe des Sensorknotens zu navigieren;
eine Mitteilung an den Sensorknoten zu übermitteln, von einem ersten Drahtlosmodus in einen zweiten Drahtlosmodus umzuschalten; und
die Firmwareaktualisierung zu übermitteln, wenn der Sensorknoten im zweiten Drahtlosmodus arbeitet.

20. System gemäß der Definition in Anspruch 19, wobei der Sensorknotenaktualisierungsmanager die Mitteilung zum Sensorknoten übermittelt, wenn der Sensorknoten im ersten Drahtlosmodus arbeitet.

21. System gemäß der Definition in Anspruch 19, wobei der Sensorknotenaktualisierungsmanager ein Objekt, welches den Sensorknoten aufweist, basierend auf Bilddaten und einer dem Objekt zugeordneten Kennung identifiziert.

22. System gemäß der Definition in Anspruch 19, wobei der Manager des unbemannten Fluggeräts eine Karte erstellt, wobei der Sensorknotenaktualisierungsmanager das unbemannte Fluggerät basierend auf der Karte navigiert.

23. Vorrichtung, umfassend:
Mittel zum Navigieren eines unbemannten Fluggeräts zu einem Sensorknoten in einem Sensorknoten Netzwerk;
Mittel zum Anweisen des Sensorknotens, von einem ersten Drahtlosmodus in einen zweiten Drahtlosmodus umzuschalten; und
Mittel zum Übermitteln einer Firmwareaktualisierung über das unbemannte Fluggerät zum Sensorknoten, wenn sich der Sensorknoten im zweiten Drahtlosmodus befindet.

24. Vorrichtung gemäß der Definition in Anspruch 23, ferner aufweisend Mittel zum Identifizieren eines Objekts, welches den Sensorknoten aufweist, wobei die Mittel zum Anweisen des Sensorknotens den Sensorknoten basierend auf der Identifizierung des Objekts anweisen.

25. Vorrichtung gemäß der Definition in 24, ferner aufweisend Mittel zum Erzeugen von Bilddaten, wobei die Mittel zum Identifizieren des Objektes das Objekt basierend auf den Bilddaten identifizieren.

Es folgen 8 Seiten Zeichnungen

Anhängende Zeichnungen

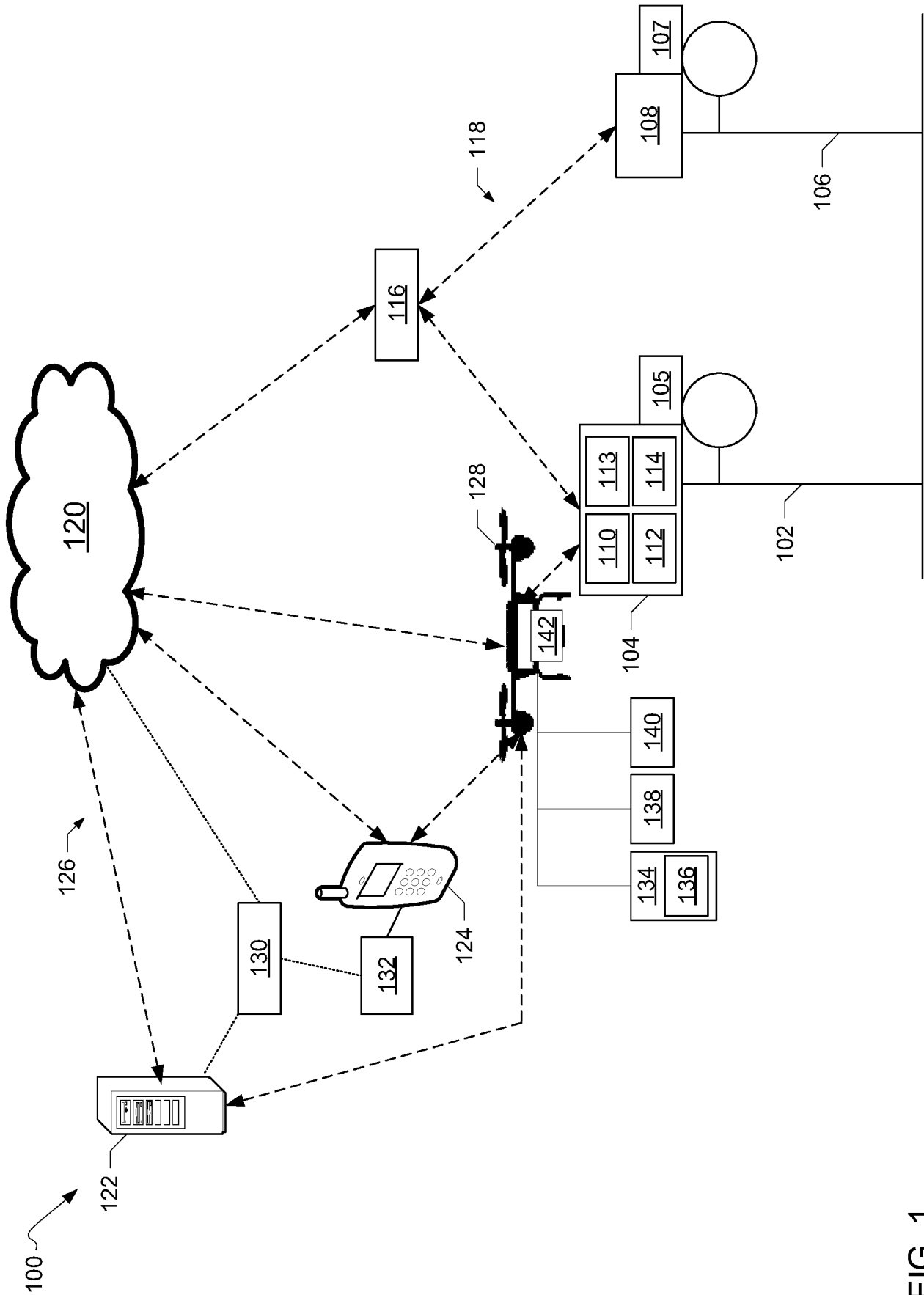


FIG. 1

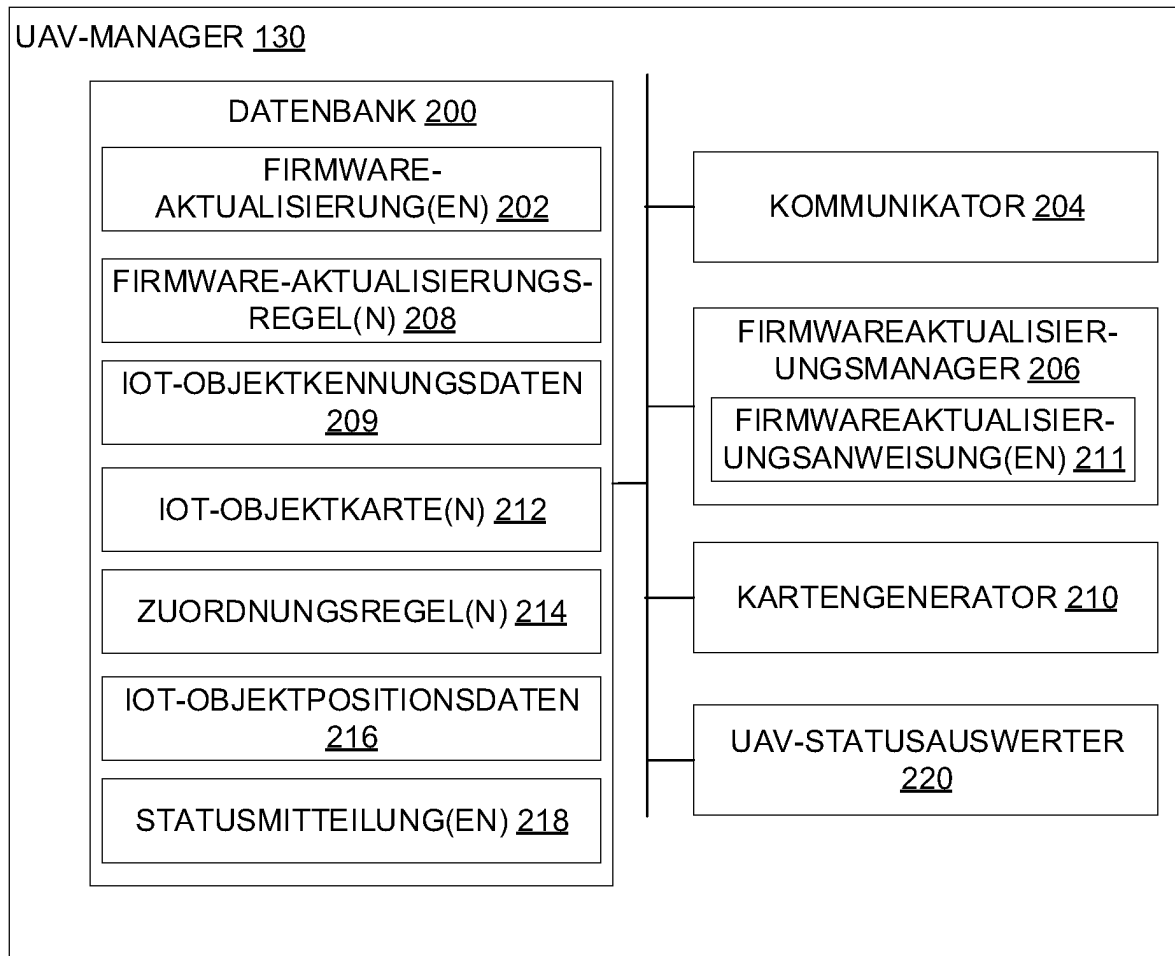


FIG. 2

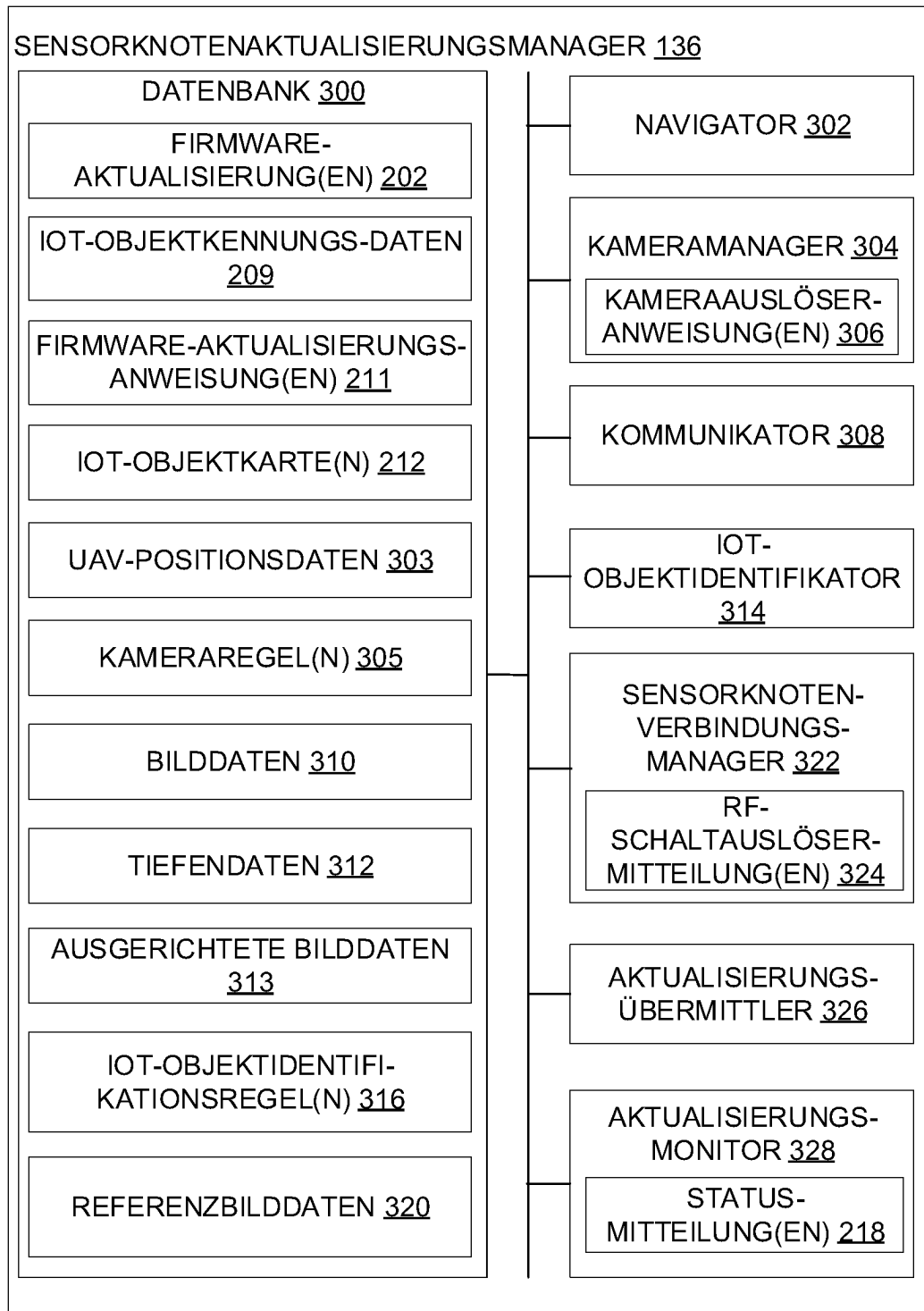


FIG. 3

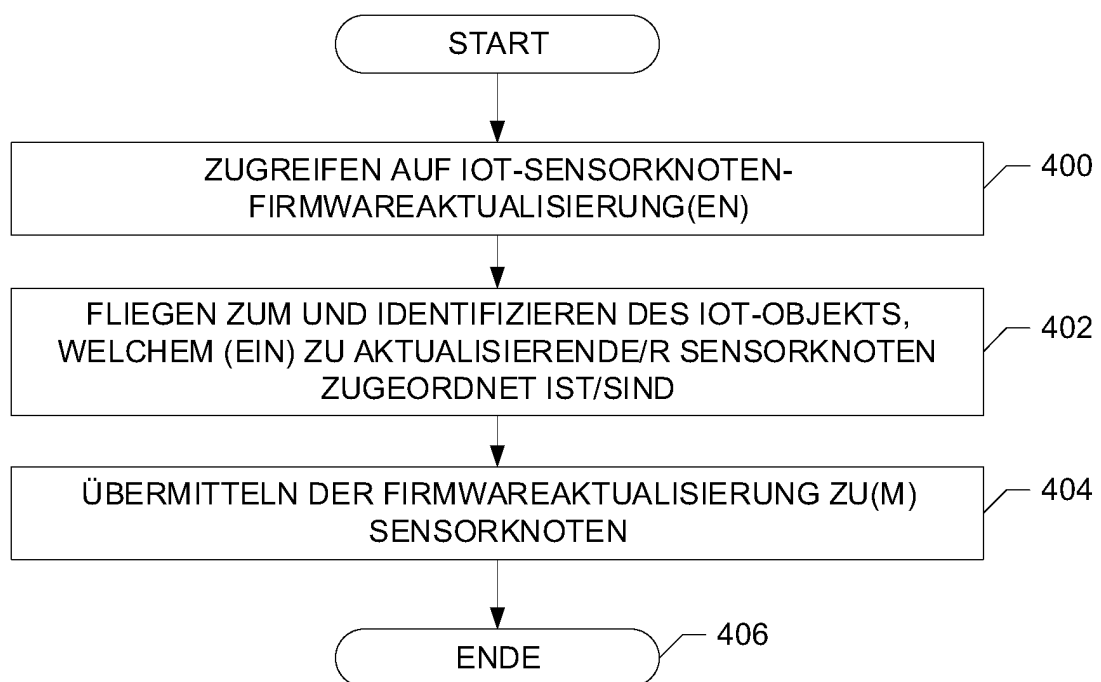


FIG. 4

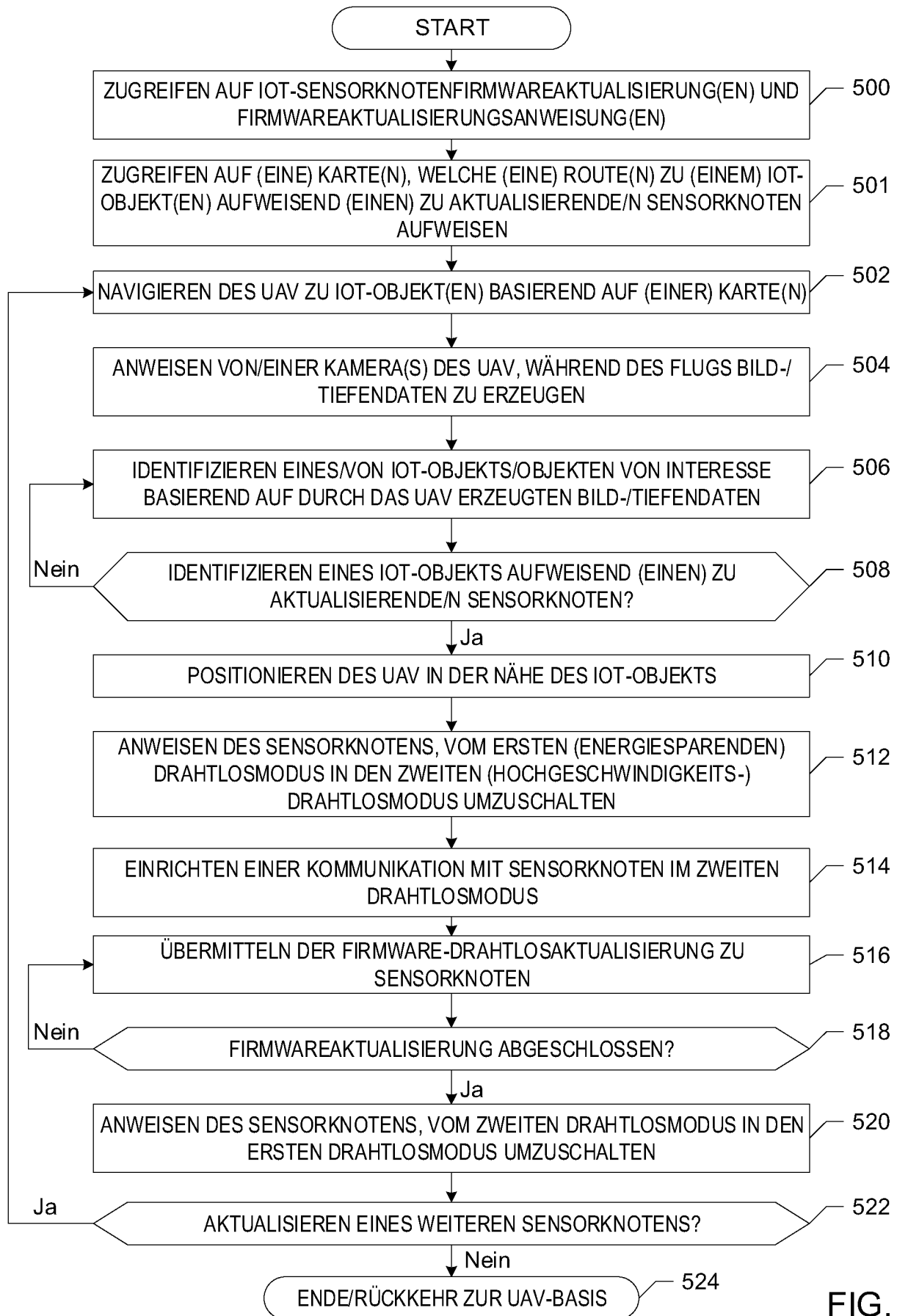


FIG. 5

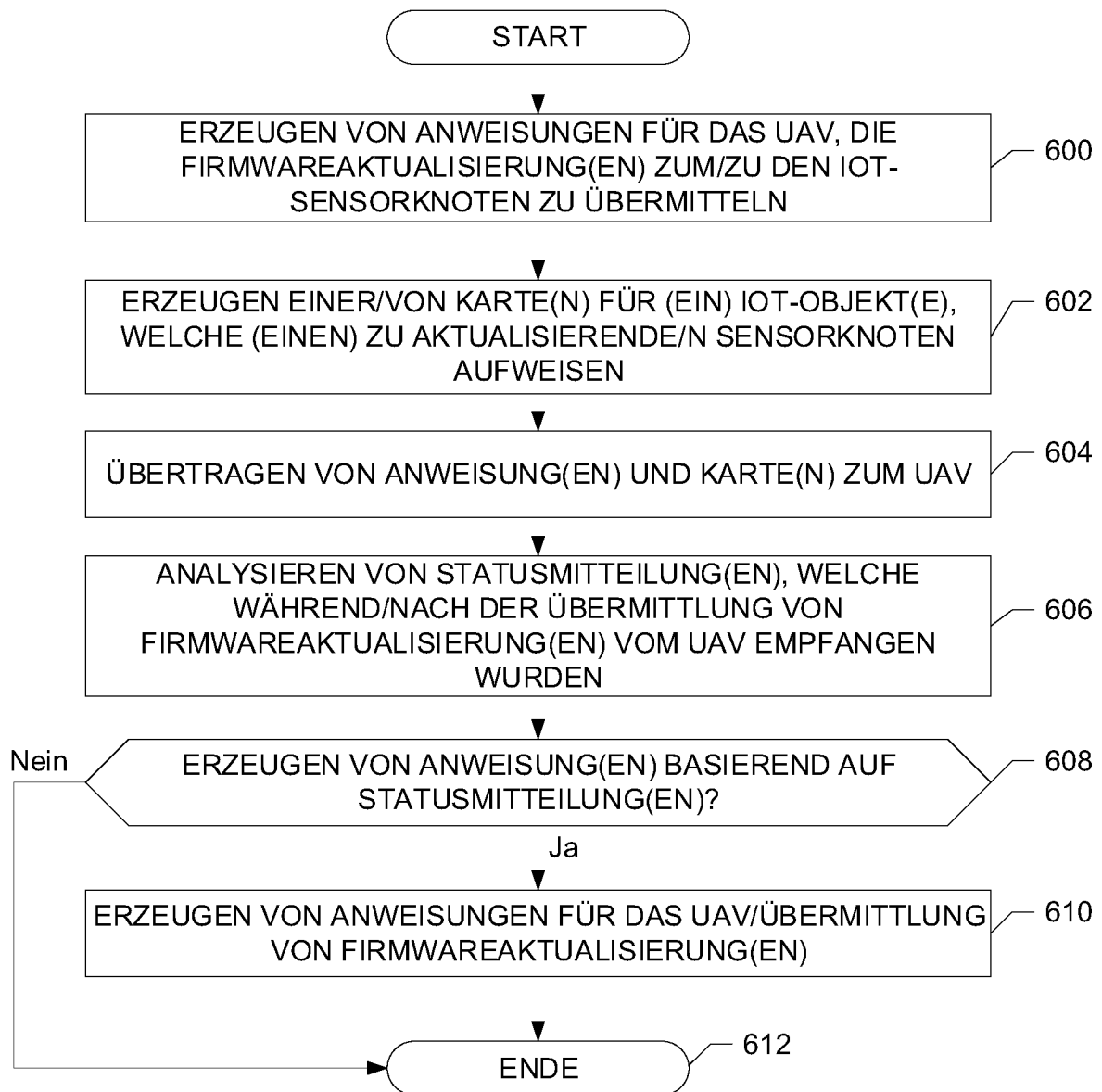


FIG. 6

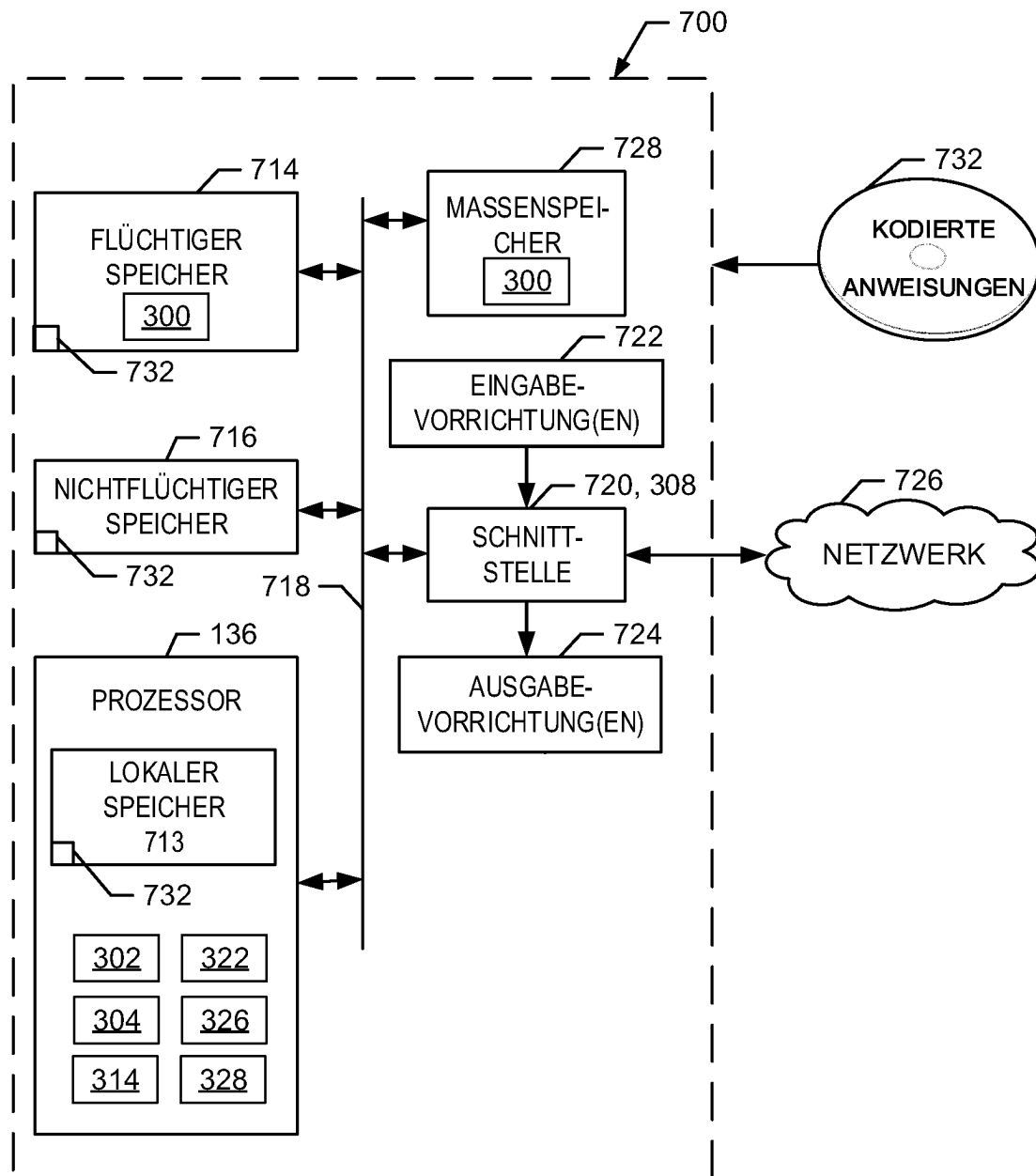


FIG. 7

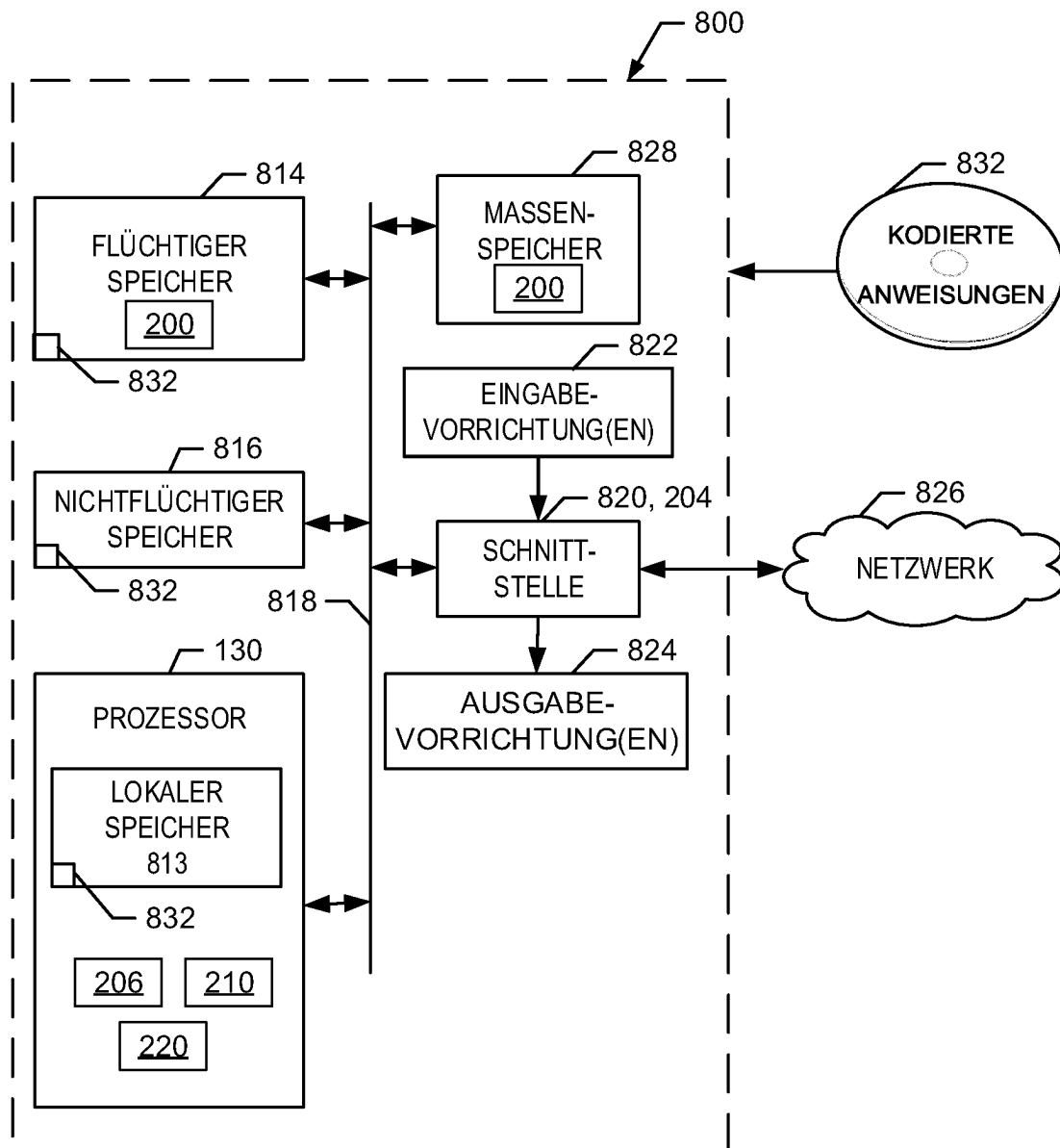


FIG. 8