



(19)
Bundesrepublik Deutschland
Deutsches Patent- und Markenamt

(10) **DE 60 2005 004 693 T2** 2009.02.12

(12) **Übersetzung der europäischen Patentschrift**

(97) **EP 1 600 893 B1**

(51) Int Cl.⁸: **G06T 7/00** (2006.01)

(21) Deutsches Aktenzeichen: **60 2005 004 693.3**

(96) Europäisches Aktenzeichen: **05 102 186.3**

(96) Europäischer Anmeldetag: **18.03.2005**

(97) Erstveröffentlichung durch das EPA: **30.11.2005**

(97) Veröffentlichungstag

der Patenterteilung beim EPA: **13.02.2008**

(47) Veröffentlichungstag im Patentblatt: **12.02.2009**

(30) Unionspriorität:

809167 23.03.2004 US

(73) Patentinhaber:

Microsoft Corp., Redmond, Wash., US

(74) Vertreter:

**Grünecker, Kinkeldey, Stockmair &
Schwanhäusser, 80802 München**

(84) Benannte Vertragsstaaten:

**AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB,
GR, HU, IE, IS, IT, LI, LT, LU, MC, NL, PL, PT, RO,
SE, SI, SK, TR**

(72) Erfinder:

**Shum, Heung-Yeung, Redmond, WA 98052, US;
Gu, Jinwei, Redmond, WA 98052, US; Lin, Stephen
S., Redmond, WA 98052, US**

(54) Bezeichnung: **RADIOMETRISCHE KALIBRIERUNG VON EINEM EINZELNEN BILD**

Anmerkung: Innerhalb von neun Monaten nach der Bekanntmachung des Hinweises auf die Erteilung des europäischen Patents kann jedermann beim Europäischen Patentamt gegen das erteilte europäische Patent Einspruch einlegen. Der Einspruch ist schriftlich einzureichen und zu begründen. Er gilt erst als eingelegt, wenn die Einspruchsgebühr entrichtet worden ist (Art. 99 (1) Europäisches Patentübereinkommen).

Die Übersetzung ist gemäß Artikel II § 3 Abs. 1 IntPatÜG 1991 vom Patentinhaber eingereicht worden. Sie wurde vom Deutschen Patent- und Markenamt inhaltlich nicht geprüft.

Beschreibung**GEBIET DER ERFINDUNG**

[0001] Verschiedene Ausführungsformen, die unten beschrieben sind, beziehen sich im Allgemeinen auf Bildverarbeitung, und spezieller gesagt, aber nicht ausschließlich, auf eine radiometrische Kalibrierung von Bildern.

HINTERGRUND DER ERFINDUNG

[0002] In manchen Anwendungen ist es wünschenswert, ein Bildaufnahmegerät (z. B. eine digitale Kamera) zu kalibrieren, so dass die Farben des Bildes (d. h. die gemessenen Farben) den tatsächlichen Farben (d. h. die Strahlung der Szene/Radianz der Szene), die durch das Bildaufnahmegerät (das hier auch einfach als Gerät bezeichnet wird) empfangen werden, genauer entsprechen.

[0003] [Fig. 1](#) (Stand der Technik) zeigt ein Beispiel eines Systems **100**, das eine digitale Kamera **102** einschließt, die eine Resonanzfunktion (response function) **104** aufweist. Im Betrieb nimmt die digitale Kamera **102** ein Bild einer Szene **106** (über deren Szeneradianz/Szenestrahlung (scene radiance)) auf, und gibt gemessene Farben aus, die verwendet werden, um ein digitales Bild **108** zu bilden. Die gemessenen Farben des digitalen Bildes **108** beziehen sich durch die Resonanzfunktion **104** auf die Szeneradianz (ebenso als die radiometrische Resonanz (radiometric response) bezeichnet). Im Allgemeinen ist die Resonanzfunktion **104** nicht linear und hängt von der Kamera ab. Des Weiteren kann die Resonanzfunktion unterschiedlich sein, selbst wenn die Kameras vom selben Modell sind.

[0004] Die Kamera **102** kann dadurch kalibriert werden, dass die „Inverse“ der Resonanzfunktion **104** gefunden wird, so dass idealerweise die gemessenen Farben auf Farben abgebildet werden, die der Szeneradianz exakt entsprechen. In einem Ansatztyp nimmt ein Benutzer ein Bild einer „Referenz“-Farbszene auf (d. h. die Szene hat Regionen einer bekannten Farbe), so dass die gemessene Farbausgabe durch die Kamera **102** mit den tatsächlichen Farben verglichen werden kann. Deshalb benötigt dieser Ansatztyp ein Bild der „Referenz“. In einem anderen Ansatztyp wird eine Vielzahl von Bildern einer Szene benötigt. In einem bestimmten Ansatz wird die Reihe von Bildern unter verschiedenen, genau bekannten Aufnahmeeinstellungen aufgenommen, wobei alle Bilder registriert werden (d. h. sie werden aufgenommen, wobei die Positionen der Kamera und der Szene unverändert bleiben).

[0005] Jedoch haben diese konventionellen Lösungen dadurch Mängel, dass in manchen Szenarien weder ein „Referenz“-Bild, das durch die Kamera aufgenommen wurde, noch eine Reihe von registrierten Bildern mit unterschiedlichen Aufnahmeeinstellungen verfügbar sein können. Z. B. kann in manchen Szenarien nur ein Bild verfügbar sein, wobei keine der Kamera- und der Aufnahmeeinstellungen, die zur Aufnahme des Bildes verwendet wurden, bekannt sind.

[0006] EP 0 562 971 A1 bezieht sich auf ein Verfahren und eine Vorrichtung zur adaptiven Charakterisierung und Kalibrierung von Farbdokumentscannern, Farbanzeigeeinheiten und Farbdruckern, und zum Bereitstellen einer eindeutigen Kalibrierungsumformung für jedes Gerät unter Verwendung eines günstigen achromatischen Sensors, der die Gerätebestandsvarianz (device population variance) dadurch minimiert, dass er ein Verfahren zum Erzeugen einer beschränkten (constrained) kanalunabhängigen Linearisierung, einer Zweck- oder Zielkurvenskalierung, um eine gewünschte Farbbalance zu erreichen, und eine Zweckkurvenskalierung (aim curve scaling) für eine Kanalsättigungskorrektur bereitstellt.

[0007] EP 1 117 070 A2 bezieht sich auf ein Diagramm und eine Methodik zur Verwendung des Diagramms, um die Farbdarstellung von digitalen Bildern zu verbessern. Des Weiteren weist das Diagramm eine Vielzahl von Farbstellen (color patches) auf, wobei jede Farbstelle Informationen einschließt, die die Zuordnung von digitalisierten Farbdaten von einem Bild erlaubt, das durch verschiedene Mittel aufgenommen wurde, um ein Profil zu konstruieren, das beim Modifizieren der Farbtonskala und Farbe des digitalen Bildes verwendbar ist, wobei die Anzahl der Farbstellen größer als 24 ist und ausgewählt wird, um nicht lineare Charakteristika der Bildaufnahme zu kompensieren.

KURZFASSUNG

[0008] Es ist die Aufgabe der Erfindung, ein verbessertes Verfahren und System zur radiometrischen Kalibrierung von Bildern und der Kalibrierung eines Bildaufnahmegerätes bereitzustellen.

[0009] Diese Aufgabe wird durch die vorliegende Erfindung, wie in den unabhängigen Ansprüchen beansprucht, gelöst.

[0010] Bevorzugte Ausführungsformen werden durch die abhängigen Ansprüche definiert.

[0011] Gemäß einem Aspekt der verschiedenen beschriebenen Ausführungsformen, werden Systeme und Verfahren zur radiometrischen Kalibrierung eines einzelnen Bildes bereitgestellt. In einem Aspekt wird ein System verwendet, um die inverse Resonanzfunktion (inverse response function) einer Kamera von einem einzelnen digitalen Bild einer Szene, in der die tatsächlichen Farben der Szene von vorneherein nicht bekannt sind, zu kalkulieren. Das System analysiert Pixel des Bildes, die einer „Kante“ zwischen zwei Farben der Szene entsprechen. Deshalb stellen diese „Kanten“-Pixel eine gemischte Farbe dar, die von diesen zwei „Komponenten“-Farben, wie sie durch die Kamera gemessen wurden, gebildet wird. Es kann gezeigt werden, dass in einem idealen Fall solch eine gemischte Farbe auf einem Linienabschnitt (Liniensegment) liegt, die ihre Komponentenfarben im RGB-Farbraum verbindet. Jedoch verursacht die Resonanzfunktion von typischen echten Kameras, dass eine gemessene gemischte Farbe in Bezug auf ihre gemessenen Komponentenfarben nicht linear ist. Gemäß diesem Aspekt ermittelt das System eine inverse Resonanzfunktion zumindest teilweise dadurch, dass: (A) geeignete Kantenpixel gefunden werden; und (B) eine Funktion ermittelt wird, die die gemessenen, gemischten Farben der Kantenpixel und deren gemessene Komponentenfarben auf lineare Verteilungen abbildet.

[0012] In einem anderen Aspekt werden Referenzdaten, die vorgegebene inverse Resonanzfunktionen von bekannten Kameras einschließen, beim Ermitteln einer inversen Resonanzfunktion verwendet. In einer Ausführungsform, die diesen Aspekt einschließt, werden Bayes'sche Abschätzungstechniken verwendet, um eine inverse Resonanzfunktion zu finden.

KURZE BESCHREIBUNG DER ZEICHNUNGEN

[0013] Nicht einschränkende und nicht erschöpfende Ausführungsformen werden mit Bezug auf die folgenden Figuren beschrieben.

[0014] [Fig. 1](#) (Stand der Technik) ist ein Blockdiagramm, das eine Kamera darstellt, die ein Bild einer Szene aufnimmt, um ein digitales Bild zu erzeugen.

[0015] [Fig. 2](#) ist ein Blockdiagramm eines Systems, das imstande ist, eine inverse Resonanzfunktion von einem einzelnen Bild gemäß einer Ausführungsform zu ermitteln.

[0016] [Fig. 3](#) ist ein Flussdiagramm, das den Funktionsablauf des Systems aus [Fig. 1](#) beim Ermitteln einer inversen Resonanzfunktion von einem einzelnen Bild gemäß einer Ausführungsform veranschaulicht.

[0017] [Fig. 4](#) ist ein Diagramm, das Kantenpixel in einem Bild veranschaulicht, die beim Ermitteln einer inversen Resonanzfunktion von einem einzelnen Bild gemäß einer Ausführungsform verwendet werden.

[0018] [Fig. 5](#) ist ein Diagramm, das die Bildstrahlen (image irradiance) der Pixel, die in [Fig. 4](#) gezeigt sind, in einem Rot-Grün-Blau-Farbraum (RGB-Farbraum) veranschaulicht.

[0019] [Fig. 6](#) ist ein Diagramm, das die gemessene Farbe der Pixel, die in [Fig. 4](#) gezeigt sind, im RGB-Farbraum veranschaulicht.

[0020] [Fig. 7](#) ist ein Diagramm, das eine gemessene Farbe eines Beispiel-Kantenpixels und die gemessenen Farben seiner Komponentenfarben im RGB-Raum veranschaulicht.

[0021] [Fig. 8](#) ist ein Diagramm, das unter Verwendung einer inversen Resonanzfunktion die resultierende Verteilung nach der Umformung der gemessenen Farben, die in [Fig. 7](#) gezeigt sind, veranschaulicht.

[0022] [Fig. 9](#) ist ein Flussdiagramm, das den Funktionsablauf, der in [Fig. 3](#) gezeigt ist, detaillierter veranschaulicht, um eine Funktion zu ermitteln, die die gemessenen, gemischten Farben gemäß einer Ausführungsform linearisiert.

[0023] [Fig. 10](#) ist ein Datenablaufdiagramm, das einen Datenablauf beim Ermitteln einer inversen Resonanzfunktion von einem einzelnen Bild gemäß einer Ausführungsform veranschaulicht.

[0024] [Fig. 11](#) ist ein Blockdiagramm, das eine Beispielcomputerumgebung darstellt, die zum Ausüben der oben genannten Ausführungsformen geeignet ist.

DETAILLIERTE BESCHREIBUNG

[0025] Verschiedene Ausführungsformen sind auf ein Verfahren und System zum Berechnen der inversen Resonanzfunktion einer Kamera von einem einzelnen digitalen Bild einer Szene gerichtet, indem die aktuellen Farben der Szene von vorneherein nicht bekannt sind. Diese Ausführungsformen analysieren Pixel des Bildes, die einer „Kante“ zwischen zwei Farben der Szene entsprechen. Die „Kantenpixel“ haben Werte, die eine gemischte Farbe darstellen, die von diesen zwei „Komponentenfarben“, wie sie durch die Kamera gemessen wurden, gebildet wird. Die Resonanzfunktion der Kamera verursacht, dass die gemessene gemischte Farbe in Bezug auf ihre gemessenen Komponentenfarben nicht linear ist. Die Ausführungsformen, die unterhalb beschrieben sind, können eine inverse Resonanzfunktion mindestens teilweise dadurch ermitteln, dass Kantenpixel gefunden werden, und anschließend eine Funktion ermittelt wird, die die gemischten Farben der Kantenpixel und die gemessenen Komponentenfarben auf lineare Verteilungen abbildet. Verschiedene Ausführungsformen werden unterhalb beschrieben.

[0026] [Fig. 2](#) ist ein Blockdiagramm, das ein Kalibrierungssystem **201** darstellt, das imstande ist, eine inverse Resonanzfunktion von einem einzelnen Bild gemäß einer Ausführungsform zu ermitteln. In diesem Beispiel schließt das System **201** einen Kantenpixeldetektor **203**, einen Farbanalysierer **205**, einen inversen Resonanzabschätzer (Schätzfunktion) **207** und einen Datenspeicher **209**, der inverse Resonanzfunktionsdaten einschließt, die von verschiedenen echten Kameras gesammelt wurden, ein. Diese Sammlung an inversen Resonanzdaten wird hier auch als Referenzdaten bezeichnet. In einer Ausführungsform umfassen die inversen Resonanzfunktionsdaten die Datenbank der Resonanzfunktionen (Database of Response Functions – DoRF) wie sie in Grossberg, M. D. und Nayar S. K., „What is the Space of Camera Response Functions“, Proc. IEEE Corp. Vision and Pattern Recognition (CVPR'03), Volume II, Juni 2003, Seiten 602 bis 609, beschrieben wird. Die oben beschriebenen Elemente des Kalibrierungssystems **201** können in manchen Ausführungsformen als Softwarekomponenten implementiert werden.

[0027] In einer Ausführungsform ist der Kantenpixeldetektor **203** so konfiguriert, dass er Kantenpixel in einem digitalen Bild **108** findet, in dem jedes der Pixel eine Region, die eine Farbe aufweist, und eine andere Region, die eine andere Farbe aufweist, zeigt (images). Jedes dieser Pixel stellt eine gemischte Farbe dar, die von zwei Farben erlangt wurde, die als Komponentenfarben der gemischten Farbe dienen.

[0028] In einer Ausführungsform ist der Farbanalysierer **205** so eingerichtet, dass er Messungen der gemischten und der Komponentenfarben der Pixel gewinnt, die durch den Kantenpixeldetektor **203** gefunden wurden. Ebenso ist in dieser Ausführungsform der inverse Resonanzabschätzer **207** eingerichtet, um eine inverse Resonanzfunktion zu generieren, die die Messungen der gemischten und der Komponentenfarben der gefundenen Pixel in eine lineare Verteilung abbildet. Eine Ausführungsform des Kalibrierungssystems **201** arbeitet, wie unterhalb in Verbindung mit [Fig. 3](#) beschrieben, um eine inverse Resonanzfunktion zu ermitteln.

[0029] [Fig. 3](#) stellt einen funktionalen Ablauf des Kalibrierungssystems **201** ([Fig. 2](#)) beim Ermitteln einer inversen Resonanzfunktion von einem einzelnen Bild gemäß einer Ausführungsform dar. Dieser beispielhafte funktionelle Ablauf wird unterhalb in Bezug auf die [Fig. 2](#) und [Fig. 3](#) beschrieben.

[0030] In einem Block **302** findet das System **201** Kantenpixel in einem digitalen Bild **108** ([Fig. 1](#)). In dieser Ausführungsform findet der Kantenpixeldetektor **203** Kantenpixel in dem digitalen Bild **108**, um einen Beobachtungssatz zu bilden. Wie vorher beschrieben sind Kantenpixel Pixel, die einer „Kante“ zwischen zwei Farben der Szene **106** entsprechen. Die Kantenpixel haben Werte, die einer gemischten Farbe entsprechen, die aus diesen zwei „Komponentenfarben“, wie sie durch die Kamera gemessen werden, gebildet wird.

[0031] Beispielhafte Kantenpixel werden in [Fig. 4](#) dargestellt, welche einen 3×3 großen Pixelausschnitt **400** (Patch) zeigt. Ein Pixel **401** in der oberen linken Ecke des Ausschnitts **400** hat eine einheitliche Farbe, was durch die Schraffur mit „diagonalen Linien“ angezeigt wird (hier als die erste Farbe bezeichnet). Zusätzlich haben die zwei Pixel in dem Ausschnitt **400** unterhalb des Pixels **401** dieselbe einheitliche erste Farbe. Im Unterschied dazu hat ein Pixel **403** in der oberen rechten Ecke des Ausschnitts **400** eine unterschiedliche, einheitliche Farbe, was durch ein Muster mit „Punkten“ gezeigt wird (hier als die zweite Farbe bezeichnet). Die zwei Pixel unterhalb des Pixels **403** haben ebenso dieselbe einheitliche zweite Farbe. Die Pixel **405**, **407** und **409** in der mittleren Spalte des Ausschnitts **400** haben Regionen mit sowohl der ersten als auch der zweiten Farbe und werden deshalb hier als Kantenpixel bezeichnet. Die erste und die zweite Farbe werden hier ebenso als

Komponentenfarben des Kantenpixels **405** bezeichnet. Das Kantenpixel **405** gibt eine „gemischte“ Farbe aus, die von einer Kombination der ersten und der zweiten Farbe gebildet wird.

[0032] In einer Ausführungsform schließt der Kantenpixeldetektor **203** einen Canny-Kantendetektor ein (siehe z. B. Canny, J., „A Computational Approach to Edge Detection“, IEEE Trans. Pattern Analysis and Machine Intelligence, Volume 8, Nr. 6, Juni 1986, Seiten 679 bis 698), um die Kantenpixel zu erkennen. In anderen Ausführungsformen können unterschiedliche Kantendetektoren verwendet werden.

[0033] In einem Block **304** wird die gemessene Farbe von jeder Region der Kantenpixel erlangt. In dieser Ausführungsform ist der Farbanalysierer **205** so eingerichtet, dass er die gemessenen, gemischten Farben von jedem Kantenpixel unter Verwendung von nicht-Kantenpixeln erlangt, die dieselben Farben, wie die Komponentenfarben in den Kantenpixeln aufweisen. Z. B. kann der Farbanalysierer **205** für das Kantenpixel **405** die gemessene Farbe des Pixels **401** für die gemessene Farbe der ersten Farbregion des Kantenpixels **405** verwenden. Gleichermaßen kann der Farbanalysierer **205** die gemessene Farbe des Pixels **403** für die gemessene Farbe der zweiten Farbregion des Kantenpixels **405** verwenden.

[0034] Des Weiteren ermittelt in dieser Ausführungsform der Farbanalysierer **205**, ob ein Kantenpixel geeignet ist, um in der Ermittlung einer inversen Resonanzfunktion verwendet zu werden. Z. B. ermittelt in einer Ausführungsform der Farbanalysierer **205**, ob: (A) die Farbvarianz (mit Bezug auf die euklidische RGB-Distanz) der ersten und der zweiten Farbe innerhalb eines festgelegten Schwellenwertes liegt (d. h. die erste und die zweite Farbe sind ausreichend einheitlich); und (B) die Durchschnittsfarbe der ersten und der zweiten Farbe haben mindestens eine festgelegte Distanz voneinander (d. h. die erste und die zweite Farbe liegen weit genug auseinander, um den Effekt des Bildrauschens zu reduzieren). Zusätzlich wird in dieser Ausführungsform eine Kantenregion (hier ebenso als Kantenfenster bezeichnet) ignoriert, die Kantenpixel enthält, die eine gemischte Farbe aufweisen, die außerhalb des Bereichs liegt, der durch die erste und die zweite Farbe begrenzt wird. In anderen Ausführungsformen kann der Farbanalysierer **205** unterschiedliche Verfahren zum Erlangen der gemessenen Farben der Komponentenfarben der Kantenpixel verwenden.

[0035] In einem Block **306** werden die gemessenen Farben der Kantenpixel erlangt. In dieser Ausführungsform ist der Farbanalysierer **205** so eingerichtet, dass die gemessene gemischte Farbe von jedem Kantenpixel erlangt wird. Es kann gezeigt werden, dass im idealen Fall eine gemischte Farbe auf einem Liniensegment liegen wird, das ihre Komponentenfarben in dem RGB-Farbraum verbindet. Ein Beispiel ist in [Fig. 5](#) gezeigt, indem die Kantenpixel aus [Fig. 4](#) im RGB-Raum eingetragen sind. Ein Punkt **501** repräsentiert die Farbe des Pixels **401** im RGB-Raum und ein Punkt **503** repräsentiert die Farbe des Pixels **403** im RGB-Raum. Diese Farben sind die Komponentenfarben der Kantenpixel in dem Ausschnitt **400** ([Fig. 4](#)). Ein Punkt **505** repräsentiert die Farbe des Kantenpixels **405** im RGB-Farbraum; ein Punkt **507** repräsentiert die Farbe des Kantenpixels **407** im RGB-Farbraum, usw. Wie es in [Fig. 5](#) gezeigt ist, liegen die Farben der Kantenpixel **405**, **407** und **409** (repräsentiert durch die Punkte **505**, **507** bzw. **509**) auf einem Liniensegment, das die Farben der Pixel **401** und **403** (repräsentiert durch die Punkte **501** bzw. **503**) verbindet.

[0036] Jedoch ist in derzeit, kommerziell verfügbaren Kameras die gemessene Farbe von einem Kantenpixel in Bezug auf dessen gemessene Komponentenfarben nicht linear. Ein Beispiel ist in [Fig. 6](#) gezeigt. Ein Punkt **601** repräsentiert die gemessene Farbe des Pixels **401** ([Fig. 4](#)) im RGB-Raum, und ein Punkt **603** repräsentiert die gemessene Farbe des Pixels **403** ([Fig. 4](#)) im RGB-Raum. Diese gemessenen Farben sind die gemessenen Komponentenfarben der Kantenpixel des Ausschnitts **400** ([Fig. 4](#)). Ein Punkt **605** repräsentiert die gemessenen Farben des Kantenpixels **405** ([Fig. 4](#)) im RGB-Farbraum; ein Punkt **607** repräsentiert die gemessene Farbe des Kantenpixels **407** im RGB-Farbraum, usw. Wie es in [Fig. 6](#) gezeigt ist, liegen die Farben der Kantenpixel **405**, **407** und **409** (repräsentiert durch die Punkte **605**, **607** bzw. **609**) auf einem Kurvensegment, das die gemessenen Farben der Pixel **401** und **403** (repräsentiert durch die Punkte **601** bzw. **603**) verbindet.

[0037] In einem Block **308** (wieder mit Bezug auf die [Fig. 1](#) und [Fig. 3](#)) wird eine Funktion ermittelt, die die gemessenen gemischten Farben und deren Komponentenfarben linearisiert (z. B. die gemessenen, nicht linearen Messungsfarben auf eine lineare Verteilung im RGB-Farbraum abbildet). In einer Ausführungsform ermittelt der Farbanalysierer **205** eine Funktion g , die die gemessenen, gemischten und gemessenen Komponentenfarben linearisiert. Des Weiteren kann in einer Ausführungsform der Farbanalysierer **205** so konfiguriert sein, dass er den Bereich (domain) und den Co-Bereich (co-domain) der Linearisierungsfunktion g normalisiert, so dass $g(0) = 0$ und $g(1) = 1$ gilt, um Skalierungsunterschiede zu berücksichtigen.

[0038] Des Weiteren verwendet in dieser Ausführungsform der Invers-Resonanzgenerator **207** anschließend die Linearisierungsfunktion g und Referenzdaten aus dem Datenspeicher **209**, um eine inverse Resonanzfunk-

tion **211** zu generieren. In einer Ausführungsform werden die Referenzdaten verwendet, um die Linearisierungsfunktion g über Intervalle von unvollständigen Farbdaten zu interpolieren und zu extrapolieren. In einer alternativen Ausführungsform kann die Linearisierungsfunktion g als die inverse Resonanzfunktion der Kamera verwendet werden, so dass der Invers-Resonanzgenerator **207** und der Datenspeicher **209** nicht implementiert werden müssen. Eine Ausführungsform des funktionalen Ablaufs zum Ermitteln der inversen Resonanzfunktion wird unterhalb in Verbindung mit [Fig. 9](#) beschrieben. Auf diese Weise kann das Kalibrierungssystem **201** vorteilhafterweise eine inverse Resonanzfunktion aus einem einzelnen Bild beschaffen. Des Weiteren müssen die tatsächlichen Szenefarben nicht vorher bekannt sein. Deshalb erfordert das Kalibrierungssystem **201** weder von der Kamera, eine inverse Resonanzfunktion zu finden, noch mehrere Bilder, die durch die Kamera aufgenommen werden, noch irgendein anderes Wissen der Aufnahmeeinstellungen, die verwendet wurden, um das Bild aufzunehmen.

[0039] Obwohl der obige funktionale Ablauf sequenziell in Verbindung mit [Fig. 3](#) beschrieben wurde, können in anderen Ausführungsformen Funktionen in unterschiedlicher Reihenfolge und/oder parallel zueinander ausgeführt werden.

[0040] Ein vereinfachtes Beispiel der Linearisierung der gemessenen Kantenpixelfarben ist grafisch in den [Fig. 7](#) und [Fig. 8](#) dargestellt. In [Fig. 7](#) repräsentiert ein Punkt **701** die gemessene Farbe von einer Komponentenfarbe von einem Kantenpixel im RGB-Raum (ebenso als $M1$ bezeichnet). Ein Punkt **703** repräsentiert die gemessene Farbe der anderen Komponentenfarbe (ebenso bezeichnet als $M2$) des Kantenpixels im RGB-Raum. Ein Punkt **705** repräsentiert die gemessene, gemischte Farbe des Kantenpixels (ebenso bezeichnet als Mp). Wie es in [Fig. 7](#) gesehen werden kann, verbindet eine Kurve die Punkte **701**, **703** und **705** aufgrund der nicht linearen Resonanzfunktion der Kamera, die verwendet wurde, um die Szene aufzunehmen. Die Kurven **706**, **708** und **710** repräsentieren die Verteilung der gemischten Farben von anderen Kantenpixels, die unterschiedliche Komponentenfarben aufweisen, als jene von der gemessenen gemischten Farbe (Mp), die durch den Punkt **705** repräsentiert wird. In einem typischen Bild würden vielmehr solcher Kurven auftreten, aber es werden nur diese wenigen in [Fig. 7](#) gezeigt, um eine Überladung der Figur zu vermeiden.

[0041] Wie oberhalb beschrieben ermittelt der Farbanalysierer **205** anschließend eine Funktion, die die gemessenen gemischten Farben und die gemessenen Komponentenfarben linearisiert. Idealerweise bildet die Funktion die gemischten Farben zugehörig zu jedem Kantenpixel (d. h. die gemessenen Komponentenfarben und die gemessene, gemischte Farbe) auf ein Liniensegment ab.

[0042] Ein Beispiel ist in [Fig. 8](#) gezeigt, wobei die Linearisierungsfunktion als Funktion g bezeichnet wird. Ein Punkt **801** repräsentiert die Farbe des Punktes **701** ([Fig. 7](#)) im RGB-Farbraum nach der Transformation durch die Linearisierungsfunktion g . Mit anderen Worten ausgedrückt, repräsentiert der Punkt **801** die Farbe der gemessenen Komponentenfarbe (d. h. $M1$) im RGB-Farbraum nach Anwendung der Linearisierungsfunktion g ; d. h. $g(M1)$. Genauso repräsentiert ein Punkt **803** die Farbe von der anderen gemessenen Komponentenfarbe (d. h. $M2$) im RGB-Farbraum nach Anwendung der Linearisierungsfunktion g ; d. h. $g(M2)$. Ein Punkt **805** repräsentiert die Farbe der gemessenen, gemischten Farbe (Mp) im RGB-Farbraum nach der Transformation durch die Linearisierungsfunktion g ; d. h. $g(Mp)$. Wie in [Fig. 8](#) gezeigt liegen die transformierten gemessenen Farben $M1$, Mp und $M2$ (repräsentiert durch die Punkte **801**, **803** bzw. **805**) auf einem Liniensegment. Ebenso werden die Kurven **706**, **708** und **710** aus [Fig. 7](#) idealerweise durch die Linearisierungsfunktion g in die Liniensegmente **806**, **808** bzw. **810** transformiert.

[0043] [Fig. 9](#) ist ein Ablaufdiagramm des funktionalen Ablaufes aus Block **308** ([Fig. 3](#)) zum Ermitteln einer Funktion, die die gemessenen, gemischten Farben gemäß einer Ausführungsform linearisiert. In anderen Ausführungsformen können unterschiedliche Ansätze verwendet werden, um eine Linearisierungsfunktion zu ermitteln. Mit Bezug auf die [Fig. 2](#) und [Fig. 3](#) schließt eine Ausführungsform des Blocks **308** einen funktionalen Ablauf wie folgt ein.

[0044] In einem Block **902** wird eine Wahrscheinlichkeitsfunktion unter Verwendung der gemessenen, gemischten Farben und deren entsprechender gemessener Komponentenfarben erlangt. In dieser Ausführungsform erlangt der Farbanalysierer **205** diese Funktion. Spezieller gesagt, erlangt in dieser Ausführungsform der Farbanalysierer **205** die Wahrscheinlichkeitsfunktion basierend auf der Distanz von jeder gemischten Farbe (Mp), die auf ein Liniensegment abgebildet wurde, das ihre entsprechenden abgebildeten Komponentenfarben ($M1$ und $M2$) verbindet. Die Werte Mp , $M1$ und $M2$ eines Kantenpixels werden hier ebenso als Kantenpixeltriple bezeichnet. Die Distanz einer abgebildeten gemischten Farbe, $d(Mp)$, zu einem Liniensegment, das die abgebildeten Komponentenfarben der gemischten Farbe verbindet, kann von einem Kantenpixeltriple unter Verwendung der unten gezeigten Gleichung 1 ermittelt werden.

$$d(Mp) = \frac{[g(M1) - g(M2)] \times [g(Mp) - g(M2)]}{|g(M1) - g(M2)|} \quad (1)$$

[0045] Wobei \times die Vektorproduktfunktion zwischen zwei Vektoren ist.

[0046] Die Summe der Distanzen von jeder abgebildeten gemischten Farbe, $D(g; \Omega)$, kann unter Verwendung der unten genannten Gleichung 2 ermittelt werden.

$$D(g; \Omega) = \sum_{\Omega} \frac{[g(M1) - g(M2)] \times [g(Mp) - g(M2)]}{g(M1) - g(M2)} \quad (2)$$

[0047] Wobei Ω eine Reihe von Kantenpixeltriplen ist, die den Kantenpixeln entsprechen, die vorher in Block 302 (Fig. 3) gefunden wurden. In manchen alternativen Ausführungsformen kann Gleichung 2 minimiert werden, um eine inverse Resonanzfunktion ohne Verwendung von Referenzdaten zu ermitteln. Z. B. kann Gleichung 2 unter Verwendung von Grenzen (constraints) minimiert werden, so dass die resultierende Gleichung sowohl gleichmäßig als auch monotonisch ist. Jedoch werden in dieser Ausführungsform Referenzdaten verwendet, um eine inverse Resonanzfunktion unter Verwendung von Bayes'schen Abschätzungstechniken zu ermitteln.

[0048] Bayes'sche Abschätzungstechniken können ein vorheriges Modell und eine Wahrscheinlichkeitsfunktion verwenden, um eine spätere Verteilung zu ermitteln. Weil die Summe der Distanzen zu minimieren ist, wird in dieser Ausführungsform die Wahrscheinlichkeitsfunktion durch das Modellieren der Wahrscheinlichkeit eines Kantenpixeltriples Ω bei gegebener inverser Resonanzfunktion g (d. h., $p(\Omega/g)$) als eine exponentielle Verteilung unter Verwendung der Gleichung 2 gebildet. Deshalb kann diese Wahrscheinlichkeitsfunktion durch die Verwendung der Gleichung 3 unten definiert werden.

$$p(\Omega|g) = \frac{1}{Z} \exp(-\lambda D(g; \Omega)) \quad (3)$$

[0049] Wobei Z eine Normalisierungskonstante ist und λ empirisch auf 10^4 gesetzt ist. In anderen Ausführungsformen können unterschiedliche Werte für λ verwendet werden.

[0050] In einem Block 904 wird ein vorheriges Modell unter Verwendung der Referenzdaten aus dem Datenspeicher 209 erlangt. In dieser Ausführungsform erlangt der inverse Resonanzgenerator 207 ein vorheriges Modell durch das Ausführen der Principal Component Analysis Transformation (PCA-Transformation) auf die zuvor genannte DoRF (d. h. die Referenzdaten) unter Verwendung von 5 Komponenten (siehe z. B. Jolliffe, I. T. „Principal Component Analysis“, Springer-Verlag, 1986). In einer Ausführungsform repräsentiert der Invers-Resonanzgenerator die Referenzdaten in Form der ersten 5 Hauptkomponenten (principal components) wie es unterhalb in Gleichung 4 gezeigt ist.

$$g = g_0 + cH \quad (4)$$

[0051] Wobei $g_0 = [g_{R0}, g_{G0}, g_{B0}]^T$ gilt, welches die mittlere inverse Resonanz ist; H ist die Matrix, deren Spalten aus den ersten $N = 5$ Eigenvektoren zusammengestellt wurden; und $c = [c_R, c_G, c_B]^T$ gilt, welches ein Koeffizientenvektor in $R^{3 \times N}$ ist, der eine inverse Resonanzfunktion $g = [g_R, g_G, g_B]^T$ repräsentiert.

[0052] Der Invers-Resonanzgenerator 207 erlangt anschließend in dieser Ausführungsform das Modell durch die inverse Resonanzfunktion durch das Bilden eines endlichen Gauß'schen Mischmodells (siehe z. B. McLachlan, G. J. und Gasford, K. E. „Mixture Models“, Marcel Dekker, 1988) aus den Referenzdaten in PCA-Form, wie in Gleichung 5 gezeigt.

$$p(g) = \sum_{i=1}^K \alpha_i \eta(g; \mu_i, \Sigma_i) \quad (5)$$

[0053] Wobei der Parameter α_i die Mischungsanteile und $\eta(g; \mu_i, \Sigma_i)$ die normalisierten Gaußwerte von g (Gleichung 8) sind. K ist in einer Ausführungsform gleich 5 gesetzt, welches empirisch unter Verwendung des Erwartungsmaximumsalgorithmus (expectation maximum (EM) algorithm) erlangt wurde (siehe z. B. Dempster, A. P., Laird, N. M. und Rubin D. B. (1977), „Maximum Likelihood from Incomplete Data via the EM algorithm (with discussion)“, JRSS (B), Volume 39, Seiten 1 bis 38).

[0054] In anderen Ausführungsformen kann das Modell aus Gleichung 4 unter Verwendung geeigneter, speziell angefertigter oder kommerziell verfügbarer PCA- und EM-Algorithmuswerkzeuge vorher berechnet werden und in dem Datenspeicher **209** abgespeichert werden. In anderen Ausführungsformen können unterschiedliche Techniken verwendet werden, um das Vormodell zu erlangen.

[0055] In einem Block **906** wird eine inverse Resonanzfunktion unter Verwendung der oben genannten Wahrscheinlichkeitsfunktion und des Vormodells erlangt. In dieser Ausführungsform erlangt der Invers-Resonanzgenerator **207** die inverse Resonanzfunktion. Die optimale inverse Resonanzfunktion g^* wird durch das Finden der maximalen Wahrscheinlichkeit der späteren Verteilung erlangt. Die spätere Verteilung wird in der Bayes'schen Abschätzung als das Produkt des vorherigen Modells und der Wahrscheinlichkeitsfunktion definiert. Deshalb kann die optimale inverse Resonanzfunktion g^* unter Verwendung der Gleichung 6, wie unterhalb gezeigt, gefunden werden, welches die maximale spätere Lösung ist (maximum a posteriori (MAP) solution).

$$g^* = \arg \max p(\Omega|g)p(g) \quad (6)$$

wobei $p(\Omega|g)$ und $p(g)$ aus den Gleichungen 3 bzw. 5 gegeben sind.

[0056] Durch das Anwenden des Logarithmus auf Gleichung 6 kann g^* ebenso als Gleichung 7 unten geschrieben werden.

$$g^* = \arg \min \lambda D(g; \Omega) - \log p(g) \quad (7)$$

welche in dieser Ausführungsform durch den Levenberg-Marquardt-Optimierungsalgorithmus berechnet wird (siehe z. B. Press, W. H., Teukolsky, S. A., Vetterling, B. T., Flannery, B. P. „Numerical Recipes in C“, Cambridge University Press, 1992), wobei die Koeffizienten von g anfänglich 0 sind. In anderen Ausführungsformen können unterschiedliche Optimierungsalgorithmen verwendet werden, um g^* zu finden. Nachdem der Optimierungsalgorithmus konvergiert, wird in einer Ausführungsform das Ergebnis sequenziell in jeder Dimension unter Verwendung einer gierigen lokalen Suche (greedy local search) verfeinert (siehe z. B., Resende, M. G. C. und Ribeiro, C. C., „Greedy randomized adaptive search procedures“, in dem Handbuch Metaheuristics „F. Glover und G. Kochenberger, eds, Kluwer Academic Publishers, Seiten 219 bis 249, 2003).

[0057] Obwohl der oben beschriebene funktionale Ablauf sequenziell in Verbindung mit [Fig. 9](#) beschrieben wurde, können in anderen Ausführungsformen Funktionen in unterschiedlicher Reihenfolge und/oder parallel zueinander durchgeführt werden.

[0058] [Fig. 10](#) ist ein Datenablaufdiagramm, das einen Datenfluss **1000** in dem Kalibrierungssystem **201** ([Fig. 2](#)) gemäß einer Ausführungsform darstellt. Bilddaten, die von einer Kamera empfangen wurden, fließen durch und werden verarbeitet, so dass Farben des Bildes gemäß einer Ausführungsform, wie folgt, angepasst werden können. Digitale Bilddaten **1003** werden durch einen Kantenpixelextraktionsprozess **1005** empfangen. In einer Ausführungsform wendet z. B. der Kantendetektor **203** des Kalibrierungssystems **201** ([Fig. 2](#)) den Kantenpixelextraktionsprozess **1005** auf die digitalen Bilddaten **1003** an, wie oben in Verbindung mit Block **302** ([Fig. 3](#)) beschrieben. Der Kantenpixelextraktionsprozess **1005** gibt Kantenpixeldaten **1007** aus, welche Kantenpixel identifizieren, die zur weiteren Verarbeitung geeignet sind (z. B. die Kantenpixel, deren Kantenfenster eine ausreichende Gleichmäßigkeit und Trennung von Komponentenfarben aufweisen).

[0059] Ein gemischter Farbextraktionsprozess **1009** wird auf Kantenpixeldaten **1007** durchgeführt. In einer Ausführungsform führt z. B. der Farbanalysierer **205** ([Fig. 2](#)) den Farbextraktionsprozess **1009** auf die Kantenpixeldaten **1007** durch, um die gemischten Farben wie oberhalb in Verbindung mit den Blöcken **304** und **306** ([Fig. 3](#)) beschrieben zu erlangen. Der gemischte Farbenextraktionsprozess **1009** gibt gemischte Farbdaten **1011** aus. In einer Ausführungsform umfassen die Kantenpixeldaten **1007** den zuvor erwähnten Satz Ω der Kantenpixeltriple.

[0060] Ein inverser Resonanz erzeugungsprozess **1013** wird anschließend auf die gemischten Farbdaten **1011** angewandt. In einer Ausführungsform führt z. B. der Invers-Resonanzgenerator **207** ([Fig. 2](#)) den Invers-Resonanzgenerationsprozess **1013** unter Verwendung der Referenzdaten **1015** durch. In manchen alternativen Ausführungsformen braucht der Invers-Resonanzgenerator **207** keine Referenzdaten **1015** verwenden. Die Referenzdaten **1015** umfassen inverse Resonanzdaten von echten Kameras und werden in einer beispielhaften Implementierung aus dem Datenspeicher **209** ([Fig. 2](#)) abgerufen. Der Invers-Resonanzgenerationsprozess **1013** wird in einer Ausführungsform, wie oberhalb in Verbindung mit Block **308** beschrieben, durch-

geführt, und gibt eine geschätzte inverse Resonanz **1017** aus. In einer Ausführungsform verwendet der Invers-Resonanzgenerationsprozess **1013** Bayes'sche Abschätzungstechniken, wie in Verbindung mit den Blöcken **902**, **904** und **906** ([Fig. 9](#)) beschrieben, um abgeschätzte inverse Resonanzen **1017** zu ermitteln. Die abgeschätzte inverse Resonanz **1017** kann anschließend verwendet werden, um digitale Bilddaten **1003** zu transformieren, um ein Bild zu produzieren, das Farben aufweist, die besser mit der Strahlung (Radianz) der aufgenommenen Szene übereinstimmen.

[0061] Obwohl der oben beschriebene Datenfluss sequenziell in Verbindung mit [Fig. 10](#) beschrieben wurde, können die Prozesse in anderen Ausführungsformen in unterschiedlicher Reihenfolge und/oder parallel zueinander auf die Daten angewandt werden. Des Weiteren können die Prozesse aus [Fig. 10](#) durch irgendein geeignetes System oder Kombinationen zusätzlich zu dem System **201** ([Fig. 2](#)) wie in der obigen Beschreibung durchgeführt werden.

[0062] Verschiedene Ausführungsformen, die oberhalb beschrieben wurden, können in Computerumgebungen des Kalibrierungssystems (z. B. System **201** aus [Fig. 2](#)) implementiert werden. Eine beispielhafte Computerumgebung, die für die Verwendung in einem Kalibrierungssystem geeignet ist, ist unterhalb in Verbindung mit [Fig. 11](#) beschrieben.

[0063] [Fig. 11](#) stellt eine allgemeine Computerumgebung **1100** dar, welche verwendet werden kann, um die Techniken und/oder Komponenten, die hier beschrieben wurden, zu implementieren. Die Computerumgebung **1100** ist nur ein Beispiel einer Computerumgebung, und ist nicht gedacht, irgendeine Einschränkung auf den Umfang der Verwendung oder Funktionalität des Computers und der Netzwerkarchitekturen vorzuschlagen. Noch sollte die Computerumgebung **1100** so interpretiert werden, als hätte sie irgendwelche Abhängigkeiten oder Erfordernisse in Bezug auf irgendeine oder Kombinationen von Komponenten, die in der exemplarischen Computerumgebung **1100** dargestellt sind. Die Computerumgebung **1100** schließt ein Allzweckcomputergerät in der Form eines Computers **1102** ein. Die Komponenten des Computers **1102** können einschließen, sind aber nicht darauf beschränkt, einen oder mehrere Prozessoren oder Prozessoreinheiten **1104**, einen Systemspeicher **1106** und einen Systembus **1108**, der verschiedene Systemkomponenten einschließlich des Prozessors **1104** mit dem Systemspeicher **1106** koppelt.

[0064] Der Systembus **1108** stellt einen oder mehrere von irgendwelchen unterschiedlichen Busstrukturtypen dar, einschließlich eines Speicherbusses oder Speichercontrollers, eines Peripheriebusses, eines beschleunigten Grafikports und eines Prozessor- oder lokalen Busses, der irgendeine einer Vielfalt von Busarchitekturen verwendet. Als Beispiel können solche Architekturen einen Industry-Standard-Architecture-Bus (ISA-BUS) einen Micro-Channel-Architecture-Bus (MCA-BUS) einen Enhanced-ISA-Bus (EISA-BUS), einen Video-Electronics-Standards-Association-Local-Bus (VESA-local-BUS), einen Peripheral-Component-Interconnect-Bus (PCI-BUS), ebenso bekannt als Mezzanine-BUS, einen PCI-Express-Bus, einen Universal-Serial-Bus (USB), einen Secure-Digital-Bus (SD-BUS) oder einen IEEE-1394-Bus (d. h. Fire-Wire-BUS) einschließen.

[0065] Der Computer **1102** kann eine Vielzahl von computerlesbaren Datenträgern einschließen. Solche Datenträger können irgendwelche verfügbaren Datenträger sein, auf die durch den Computer **1102** zugegriffen werden kann, und die sowohl flüchtige als auch nicht-flüchtige Datenträger, entfernbare und nicht-entfern timer Datenträger einschließen. Der Systemspeicher **1106** schließt computerlesbare Datenträger in der Form von flüchtigem Speicher, wie z. B. Random Access Memory (RAM – Arbeitsspeicher) **1110**; und/oder nicht-flüchtigem Speicher, wie z. B. Read Only Memory (ROM – nur Lesespeicher) **1112** oder Flash-RAM ein. Ein Basic-Input/Output-System (BIOS) **1114**, das die Basisroutinen enthält, die helfen, Informationen zwischen Elementen des Computers **1102** z. B. während des Hochfahrens zu übertragen, ist in dem ROM **1112** oder dem Flash-RAM gespeichert. Der RAM **1110** enthält üblicherweise Daten und/oder Programmmodule, auf die durch die Prozessoreinheit **1104** sofort zugegriffen werden kann, und/oder auf denen derzeit durch die Prozessoreinheit **1104** gearbeitet wird.

[0066] Der Computer **1102** kann ebenso andere entfern timer/nicht-entfern timer, flüchtige/nicht-flüchtige Computerdatenspeicherträger einschließen. Als Beispiel stellt die [Fig. 11](#) ein Festplattenlaufwerk **1116** zum Lesen von und Schreiben zu eines nicht-entfern timer, nichtflüchtigen magnetischen Datenträgers (nicht gezeigt), ein magnetisches Disklaufwerk **1118** zum Lesen von und Schreiben zu einer entfernbaren, nicht-flüchtigen magnetischen Disk **1120** (z. B. eine „Floppy-Disk“) und ein optisches Disklaufwerk **1122** zum Lesen von und/oder Schreiben zu einer entfernbaren, nicht-flüchtigen optischen Disk **1124**, wie z. B. einer CD-ROM, DVD-ROM oder anderen optischen Datenträgern, dar. Das Festplattenlaufwerk **1116**, magnetische Disklaufwerk **1118** und optische Disklaufwerk **1122** sind jeweils mit dem Systembus **1108** durch eine oder mehrere Datenträgerschnittstellen **1125** verbunden. Alternativ dazu können das Festplattenlaufwerk **1116**, magnetische Disklaufwerk **1118**

und optische Disklaufwerk **1122** über eine oder mehrere Schnittstellen (nicht gezeigt) mit dem Systembus **1108** verbunden sein.

[0067] Die Disklaufwerke und ihre zugehörigen computerlesbaren Datenträger stellen nichtflüchtigen Speicher für computerlesbare Instruktionen, Datenstrukturen, Programmmodule und andere Daten für den Computer **1102** bereit. Obwohl das Beispiel eine Festplatte **1116**, eine entfernbare magnetische Disk **1120** und eine entfernbare optische Disk **1124** darstellt, wird es begrüßt, dass andere Typen von computerlesbaren Datenträgern, welche Daten speichern können, auf die durch einen Computer zugegriffen werden kann, wie z. B. magnetische Kassetten oder andere magnetische Speichergeräte, Flash-Memory-Karten, CD-ROM, Digital Versatile Discs (DVD) oder andere optische Speicher, Random Access Memories (RAM), Read Only Memories (ROM), elektrische, löschbare, programmierbare Nur-Lesespeicher (electrically erasable programmable read only memory – EEPROM) und ähnliche, ebenso verwendet werden können, um das beispielhafte Computersystem und die Umgebung zu implementieren.

[0068] Irgendeine Anzahl von Programmmodulen kann auf der Festplatte **1116**, der magnetischen Disk **1120**, der optischen Disk **1124**, ROM **1112** und/oder RAM **1110** gespeichert werden, zum Beispiel einschließlich des Betriebssystems **1126**, eines oder mehrerer Anwendungsprogramme **1128**, anderer Programmmodule **1130** und Programmdateien **1132**. Jedes solcher Betriebssysteme **1126**, eines oder mehrerer Anwendungsprogramme **1128**, anderer Programmmodule **1130** und Programmdateien **1132** (oder irgendeine Kombination davon) kann alle oder einen Teil der lokal liegenden (resident) Komponenten, die das verteilte Dateisystem unterstützen, implementieren.

[0069] Ein Benutzer kann Befehle und Informationen in den Computer **1102** über Eingabegeräte, wie z. B. eine Tastatur **1134** und ein Zeigergerät **1136** (z. B. eine „Maus“), eingeben. Andere Eingabegeräte **1138** (nicht speziell gezeigt) können z. B. ein Mikrofon, einen Joystick, ein Gamepad, eine Satellitenschüssel, einen seriellen Anschluss, eine Scanner und/oder ähnliches einschließen. Diese und andere Eingabegeräte werden mit deressoreinheit **1104** über Eingabe-/Ausgabeschnittstellen **1140** verbunden, die mit dem Systembus **1108** gekoppelt sind, können aber durch andere Schnittstellen und Busstrukturen, wie z. B. einem Parallelanschluss, einem Gameport oder einem Universal Serial Bus (USB) verbunden werden.

[0070] Ein Monitor **1142** oder anderer Typ von Anzeigegerät kann ebenso mit dem Systembus **1108** über eine Schnittstelle verbunden sein, wie z. B. einem Videoadapter **1144**. Zusätzlich zu dem Monitor **1142** können andere Ausgabegeräte Komponenten wie z. B. Lautsprecher (nicht gezeigt) und Drucker **1146** einschließen, welche über I/O-Schnittstellen **1140** mit dem Computer **1102** verbunden sein können.

[0071] Der Computer **1102** kann in einer Netzwerkumgebung unter Verwendung logischer Verbindungen zu einem oder mehreren Remote-Computern arbeiten, wie z. B. dem Remote-Computergerät **1148**. Als Beispiel kann der Remote-Computer **1148** ein PC, portabler Computer, ein Server, ein Router, ein Netzwerkcomputer, ein Peer-Gerät oder anderer bekannter Netzwerkknoten oder ähnliches sein. Das Remote-Computergerät **1148** wird als ein portabler Computer dargestellt, der viele oder alle der Elemente und Merkmale, die oben mit Bezug auf Computer **1102** beschrieben sind, einschließen. Alternativ kann der Computer **1102** ebenso in einer Nicht-Netzwerkumgebung arbeiten.

[0072] Logische Verbindungen zwischen dem Computer **1102** und dem Remote-Computer **1148** sind als ein Local Area Network (LAN) **1150** und ein allgemeines Wide Area Network (WAN) **1152** gezeigt. Solche Netzwerkumgebungen sind alltäglich in Büros, unternehmensweiten Computernetzwerken, Intranets und dem Internet.

[0073] Wenn er in einer LAN-Netzwerkumgebung implementiert ist, ist der Computer **1102** mit dem lokalen Netzwerk **1150** über eine Netzwerkschnittstelle oder -adapter **1154** verbunden. Wenn er in einer WAN-Netzwerkumgebung implementiert ist, schließt der Computer **1102** üblicherweise ein Modem **1156** oder andere Mittel zum Herstellen von Datenübertragungsverbindungen über das breitbandige (wide) Netzwerk **1152** ein. Das Modem **1156**, welches intern oder extern zu dem Computer **1102** sein kann, kann mit dem Systembus **1108** über I/O-Schnittstellen **1140** oder andere geeignete Mechanismen verbunden sein. Es gilt als verstanden, dass die dargestellten Netzwerkverbindungen Beispiele sind, und das andere Mittel zum Herstellen mindestens einer Datenübertragungsverbindung zwischen den Computern **1102** und **1148** eingesetzt werden können.

[0074] In einer Netzwerkumgebung, wie z. B. die, die in Verbindung mit Computerumgebung **1100** dargestellt ist, können Programmmodule, die mit Bezug auf Computer **1102** gezeigt sind, oder Teile davon, in einem entfernten (remote) Datenspeichergerät gespeichert sein. Als Beispiel liegen die Remote-Anwendungsprogram-

me **1158** auf einem Speichergerät des Remote-Computers **1148**. Zum Zweck der Darstellung sind die Anwendungen oder Programme und anderen ausführbaren Programmkomponenten, wie z. B. das Betriebssystem, hier als getrennte Blöcke dargestellt, obwohl es anerkannt werden kann, dass solche Programme und Komponenten zu verschiedenen Zeiten in verschiedenen Speicherkomponenten des Computergerätes **1102** liegen können und durch mindestens einen Datenprozessor des Computers verarbeitet werden können.

[0075] Verschiedene Module und Techniken können hier im allgemeinen Kontext von computerausführbaren Instruktionen beschrieben werden, wie z. B. Programmmodulen, die durch einen oder mehrere Computer oder andere Geräte ausgeführt werden. Im Allgemeinen schließen Programmmodule Routinen, Programme, Objekte, Komponenten, Datenstrukturen etc. zum Ausführen bestimmter Aufgaben oder Implementieren bestimmter abstrakter Datentypen ein. Diese Programmmodule und ähnliches können als alternativer Code ausgeführt werden, oder können heruntergeladen und ausgeführt werden, wie z. B. in einer virtuellen Maschine oder anderen Just-in-time-Kompilierungsausführungsumgebung. Üblicherweise kann die Funktionalität der Programmmodule kombiniert oder verteilt sein, wie es in verschiedenen Ausführungsformen gewünscht ist.

[0076] Eine Implementierung dieser Module und Techniken kann auf irgendeiner Form von computerlesbarem Datenträger gespeichert werden oder damit übermittelt werden. Computerlesbare Datenträger können irgendwelche verfügbaren Datenträger sein, auf die durch einen Computer zugegriffen werden kann. Z. B. und nicht als Einschränkung gedacht, können computerlesbare Datenträger „Computerspeicherdatenträger“ und „Datenübertragungsdatenträger“ umfassen.

[0077] „Computerspeicherdatenträger“ schließen flüchtige und nicht-flüchtige, entfernbare und nicht-entfernbar Datenträger ein, die durch irgendein Verfahren oder eine Technologie zum Speichern von Informationen implementiert sind, wie z. B. Computer-lesbare Instruktionen, Datenstrukturen, Programmmodule oder andere Daten. Computerspeicherdatenträger schließen ein, sind aber nicht darauf begrenzt: RAM, ROM, EEPROM, Flash Memory oder andere Speichertechnologie, CD-ROM, Digital Versatile Discs (DVD) oder andere optische Speicher, magnetische Kassetten, magnetische Bänder, magnetische Diskspeicher oder andere magnetische Speichergeräte oder irgendein anderer Datenträger, der verwendet werden kann, um die gewünschten Informationen zu speichern, und auf den durch einen Computer zugegriffen werden kann.

[0078] „Datenübertragungsdatenträger“ verkörpern üblicherweise Computer-lesbare Instruktionen, Datenstrukturen, Programmmodule oder andere Daten in einem modulierten Datensignal, wie z. B. einer Trägerwelle oder anderem Transportmechanismus. Datenübertragungsdatenträger schließen ebenso irgendwelche Informationslieferdatenträger ein. Der Begriff „moduliertes Datensignal“ bedeutet ein Signal, das eine oder mehrere seiner Charakteristiken in solch einer Weise gesetzt oder verändert hat, dass es Informationen in dem Signal codiert. Als ein nicht einschränkendes Beispiel schließen Datenübertragungsdatenträger verkabelte Datenträger, wie z. B. ein verkabeltes Netzwerk oder direkt verkabelte Verbindungen, und kabellose Datenträger, wie z. B. akustische, RF-, Infrarot-, und andere kabellose Datenträger, ein. Kombinationen von irgendwelchen oben genannten sind ebenso im Umfang von den Computer-lesbaren Datenträgern eingeschlossen.

[0079] Durch die gesamte Beschreibung hindurch wurde auf „eine Ausführungsform“, „Ausführungsform“ oder „eine exemplarische Ausführungsform“ Bezug genommen, was bedeutet, dass ein bestimmtes beschriebenes Merkmal, Struktur oder Charakteristik in mindestens einer Ausführungsform der vorliegenden Erfindung eingeschlossen ist. Deshalb kann die Verwendung solcher Begriffe auf mehr als eine Ausführungsform verweisen. Des Weiteren können die beschriebenen Merkmale, Strukturen oder Charakteristiken in irgendeiner geeigneten Weise in einer oder mehreren Ausführungsformen kombiniert werden.

[0080] Eine Person, die in der entsprechenden Fachwelt ausgebildet ist, wird anerkennen, dass die Erfindung jedoch ohne eines oder mehrere der bestimmten Details, oder mit anderen Verfahren, Ressourcen, Materialien, etc. durchgeführt werden kann. In anderen Fällen sind gut bekannte Strukturen, Ressourcen oder Funktionen nicht gezeigt oder detailliert beschrieben worden, um lediglich zu verhindern, dass Aspekte der Erfindung unklar werden.

[0081] Während beispielhafte Ausführungsformen und Anwendungen dargestellt wurden und beschrieben worden sind, gilt es als verstanden, dass die Erfindung nicht auf die exakte Konfiguration und Ressourcen, die oberhalb beschrieben wurden, beschränkt ist. Verschiedene Modifikationen, Veränderungen und Variationen, die jenen, die in der Fachwelt ausgebildet sind, bekannt sind, können in der Anordnung, Funktion und den Details des Verfahrens und Systems der vorliegenden Erfindung, die hier offenbart wurde, gemacht werden, ohne von dem Umfang der beanspruchten Erfindung abzuweichen.

Patentansprüche

1. Verfahren zum Ermitteln einer inversen Resonanzfunktion (inverse response function) einer Kamera, wobei das Verfahren umfasst:
Finden (302) eines ersten Pixels (405) in einem Ausgabebild der Kamera, in dem das erste Pixel eine erste Region, die eine erste Farbe aufweist, und eine zweite Region, die eine zweite Farbe aufweist, abbildet, wobei das erste Pixel eine vermischte Farbe darstellt, die sich von der ersten und der zweiten Farbe ableitet, und wobei die erste und die zweite Farbe als Komponentenfarben der vermischten Farbe dienen;
Erlangen (304) der Kameramesswerte von der ersten und der zweiten Farbe;
Erlangen (306) des Kameramesswertes von der vermischten Farbe; und
Finden (308) einer Funktion, die die Messwerte der ersten, zweiten und vermischten Farbe auf eine lineare Verteilung in einem Farbraum (color space) abbildet (map).
2. Verfahren nach Anspruch 1, das des Weiteren umfasst:
Finden einer Vielzahl von Pixeln (405, 407, 409) in einem Ausgabebild, in dem jedes Pixel der Vielzahl von Pixeln zwei Regionen mit unterschiedlichen Farben abbildet und eine vermischte Farbe darstellt, die sich von den unterschiedlichen Farben ableitet, wobei die unterschiedlichen Farben von jedem Pixel als Komponentenfarben der vermischten Farbe dieses Pixels dienen;
Erlangen der Kameramesswerte von den verschiedenen Farben von jedem Pixel der Vielzahl von Pixeln;
Erlangen des Kameramesswertes von den vermischten Farben der Vielzahl von Pixeln; und
Finden einer Funktion, die die Messwerte der Farben von dem ersten Pixel und von der Vielzahl von Pixeln auf eine lineare Verteilung in dem Farbraum abbildet.
3. Verfahren nach Anspruch 1, wobei der Messwert der ersten Farbe von einem zweiten Pixel erlangt wird, das nur die erste Farbe abbildet.
4. Verfahren nach Anspruch 1, wobei das zweite Pixel angrenzend an das erste Pixel ist.
5. Verfahren nach Anspruch 1, wobei das Finden einer Funktion, die die Messwerte der ersten, zweiten und vermischten Farbe auf eine lineare Funktion abbildet, des Weiteren das Ermitteln einer Distanzfunktion umfasst, die eine Summe von Distanzen von jedem abgebildeten Messwert einer vermischten Farbe zu einem Liniensegment, das die abgebildeten Komponentenfarben in dem Farbraum der vermischten Farbe verbindet, minimiert.
6. Verfahren nach Anspruch 2, wobei das Finden einer Funktion, die die Messwerte der Farben von dem ersten Pixel und der Vielzahl von Pixeln auf eine lineare Verteilung abbildet, des Weiteren das Ermitteln der Funktion mit einer Abhängigkeit auf vorgegebene Resonanzfunktionen von bekannten Kameras umfasst.
7. Verfahren nach Anspruch 1, das des Weiteren das Verwenden eines Bayesischen Abschätzungsalgorithmus (Bayesian estimation algorithm) zum Ermitteln der Funktion umfasst.
8. Verfahren nach Anspruch 6, das des Weiteren das Modellieren der vorgegebenen Resonanzfunktionen als ein Gauß'sches Vermischungsmodell (Gaussian mixture model) umfasst.
9. Verfahren nach Anspruch 5, das des Weiteren das Einbeziehen der Distanzfunktion in eine exponentielle Verteilungsfunktion umfasst.
10. Verfahren nach Anspruch 2, das des Weiteren das Finden einer Lösung nach dem Maximum a posteriori, MAP, umfasst, die als das Produkt eines vorherigen Modells und einer Wahrscheinlichkeitsfunktion formuliert ist, wobei das vorherige Modell ein Gauß'sches Vermischungsmodell ist, das sich von den vorgegebenen Resonanzfunktionen ableitet, wobei die Wahrscheinlichkeitsfunktion eine exponentielle Verteilungsfunktion ist, die sich von Distanzen von jedem abgebildeten Messwert einer vermischten Farbe zu einem Liniensegment, das die abgebildeten Komponentenfarben der vermischten Farbe in dem Farbraum verbindet, ableitet, und wobei sich die inverse Resonanzfunktion von der MAP-Lösung ableitet.
11. Verfahren nach Anspruch 1, wobei mindestens manche der Farben einer Szene des Ausgabebildes nicht vorher (a priori) bekannt sind.
12. Maschinenlesbarer Datenträger, der Instruktionen zum Durchführen des Verfahrens nach Anspruch 1 aufweist.

13. System, das umfasst:

einen Grenzpixeldetektor (edge Pixel detector) (**203**), um eine Vielzahl von Pixeln (**405**, **407**, **409**) in einem digitalen Bild (**108**) zu finden, in dem jedes Pixel eine erste Region, die eine erste Farbe aufweist, und eine zweite Region, die eine zweite Farbe aufweist, abbildet, wobei dieses Pixel eine vermischte Farbe darstellt, die sich von der ersten und von der zweiten Farbe ableitet, und wobei die erste und die zweite Farbe als Komponentenfarben der vermischten Farbe dieses Pixels dienen;

ein Farbanalysator (**205**), der in wirksamer Weise mit dem Grenzpixeldetektor gekoppelt ist, wobei der Farbanalysator eingerichtet ist zum Erlangen von Messwerten der vermischten und der Komponentenfarben der Vielzahl von Pixeln; und

ein umgekehrter Resonanzgenerator (inverse response generator) (**207**), um eine inverse Resonanzfunktion (**211**) zu generieren, die die Messwerte der vermischten und der Komponentenfarben der Vielzahl von Pixeln auf eine lineare Verteilung in einem Farbraum abbildet.

14. System nach Anspruch 13, wobei der inverse Resonanzgenerator eingerichtet ist zum Ermitteln einer Distanzfunktion, die, für die Vielzahl von Pixeln, eine Summe an Distanzen von jedem abgebildeten Messwert einer vermischten Farbe zu einem Liniensegment, das die abgebildeten Komponentenfarben der vermischten Farbe in dem Farbraum verbindet, minimiert.

15. System nach Anspruch 13, das des Weiteren einen Datenspeicher (**209**) umfasst, um Referenzdaten, die vorgegebene Resonanzfunktionen von bekannten Kameras umfassen, zu enthalten, und wobei der inverse Resonanzgenerator eingerichtet ist zum Ermitteln der inversen Resonanzfunktion in Abhängigkeit von den Referenzdaten des Datenspeichers.

16. System nach Anspruch 15, wobei der inverse Resonanzgenerator des Weiteren eingerichtet ist zum Verwenden eines Bayesischen Abschätzungsalgorithmus, um die inverse Resonanzfunktion zu ermitteln.

17. System nach Anspruch 15, wobei der inverse Resonanzgenerator des Weiteren eingerichtet ist zum Modellieren der vorgegebenen Resonanzfunktionen als ein Gauß'sches Vermischungsmodell.

18. System nach Anspruch 14, wobei der inverse Resonanzgenerator des Weiteren eingerichtet ist zum Einbeziehen der Distanzfunktion in eine exponentielle Verteilungsfunktion.

19. System nach Anspruch 13, wobei der inverse Resonanzgenerator des Weiteren eingerichtet ist zum Ermitteln einer Lösung nach dem Maximum a posteriori, MAP, als das Produkt eines vorherigen Modells und einer Wahrscheinlichkeitsfunktion, wobei das vorherige Modell ein Gauß'sches Vermischungsmodell ist, das sich von vorgegebenen Resonanzfunktionen ableitet, wobei die Wahrscheinlichkeitsfunktion eine exponentielle Verteilungsfunktion ist, die sich von Distanzen von jedem abgebildeten Messwert einer vermischten Farbe zu einem Liniensegment, das die abgebildeten Komponentenfarben der vermischten Farbe in dem Farbraum verbindet, ableitet, und wobei sich die inverse Resonanzfunktion von der MAP-Lösung ableitet.

20. System nach Anspruch 19, wobei die MAP-Lösung als die inverse Resonanzfunktion dient.

21. System nach Anspruch 19, wobei der inverse Funktionsgenerator eingerichtet ist zum Ermitteln der MAP-Funktion unter Verwendung eines Levenberg-Marquardt-Optimierungsverfahrens.

22. Maschinenlesbarer Datenträger, der Komponenten aufweist zum Implementieren des Systems, wie es in Anspruch 13 wiedergegeben ist.

23. Maschinenlesbarer Datenträger, der Komponenten aufweist, die umfassen:

Mittel zum Finden einer Vielzahl von Pixeln in einem Ausgabebild, in dem jedes Pixel der Vielzahl von Pixeln zwei Regionen mit unterschiedlichen Farben abbildet und eine vermischte Farbe darstellt, die sich von den unterschiedlichen Farben ableitet, und wobei die unterschiedlichen Farben von jedem Pixel als Komponentenfarben von der vermischten Farbe dieses Pixels dienen;

Mittel zum Erlangen von Messwerten von den verschiedenen Farben von jedem Pixel aus der Vielzahl von Pixeln;

Mittel zum Erlangen von Messwerten der vermischten Farben der Vielzahl von Pixeln; und

Mittel zum Ermitteln einer inversen Resonanzfunktion, die die Messwerte der Farben von der Vielzahl von Pixeln auf eine lineare Verteilung in dem Farbraum abbildet.

24. Maschinenlesbarer Datenträger nach Anspruch 23, wobei die Mittel zum Ermitteln einer inversen Re-

sonanzfunktion des Weiteren Mittel zum Generieren einer Distanzfunktion umfassen, die eine Summe von Distanzen von jedem abgebildeten Messwert einer vermischten Farbe zu einem Liniensegment, das die abgebildeten Komponentenfarben der vermischten Farbe in dem Farbraum verbindet, minimiert.

25. Maschinenlesbarer Datenträger nach Anspruch 23, wobei die Mittel zum Ermitteln einer inversen Resonanzfunktion des Weiteren eingerichtet sind zum Ermitteln der Funktion in Abhängigkeit von vorgegebenen Resonanzfunktionen von bekannten Kameras.

26. Maschinenlesbarer Datenträger nach Anspruch 25, der des Weiteren Mittel zum Modellieren der vorgegebenen Resonanzfunktionen als ein Gauß'sches Vermischungsmodell umfasst.

27. Maschinenlesbarer Datenträger nach Anspruch 24, der des Weiteren Mittel zum Einbeziehen der Distanzfunktion in eine exponentielle Verteilungsfunktion umfasst.

28. Maschinenlesbarer Datenträger nach Anspruch 23, der des Weiteren Mittel zum Finden einer Lösung nach dem Maximum a posteriori, MAP, als das Produkt eines vorherigen Modells und einer Wahrscheinlichkeitsfunktion umfasst, wobei das vorherige Modell ein Gauß'sches Vermischungsmodell ist, das sich von vorgegebenen Resonanzfunktionen ableitet, wobei die Wahrscheinlichkeitsfunktion eine exponentielle Verteilungsfunktion ist, die sich von Distanzen von jedem abgebildeten Messwert einer vermischten Farbe zu einem Liniensegment, das die abgebildeten Komponentenfarben der vermischten Farbe in dem Farbraum verbindet, ableitet, und wobei sich die inverse Resonanzfunktion von der MAP-Lösung ableitet.

29. Maschinenlesbarer Datenträger nach Anspruch 28, wobei die MAP-Lösung als die inverse Resonanzfunktion dient.

30. Maschinenlesbarer Datenträger nach Anspruch 28, wobei die Mittel zum Finden einer MAP-Lösung ein Levenberg-Marquardt-Optimierungsverfahren zum Finden der MAP-Lösung verwenden.

Es folgen 6 Blatt Zeichnungen

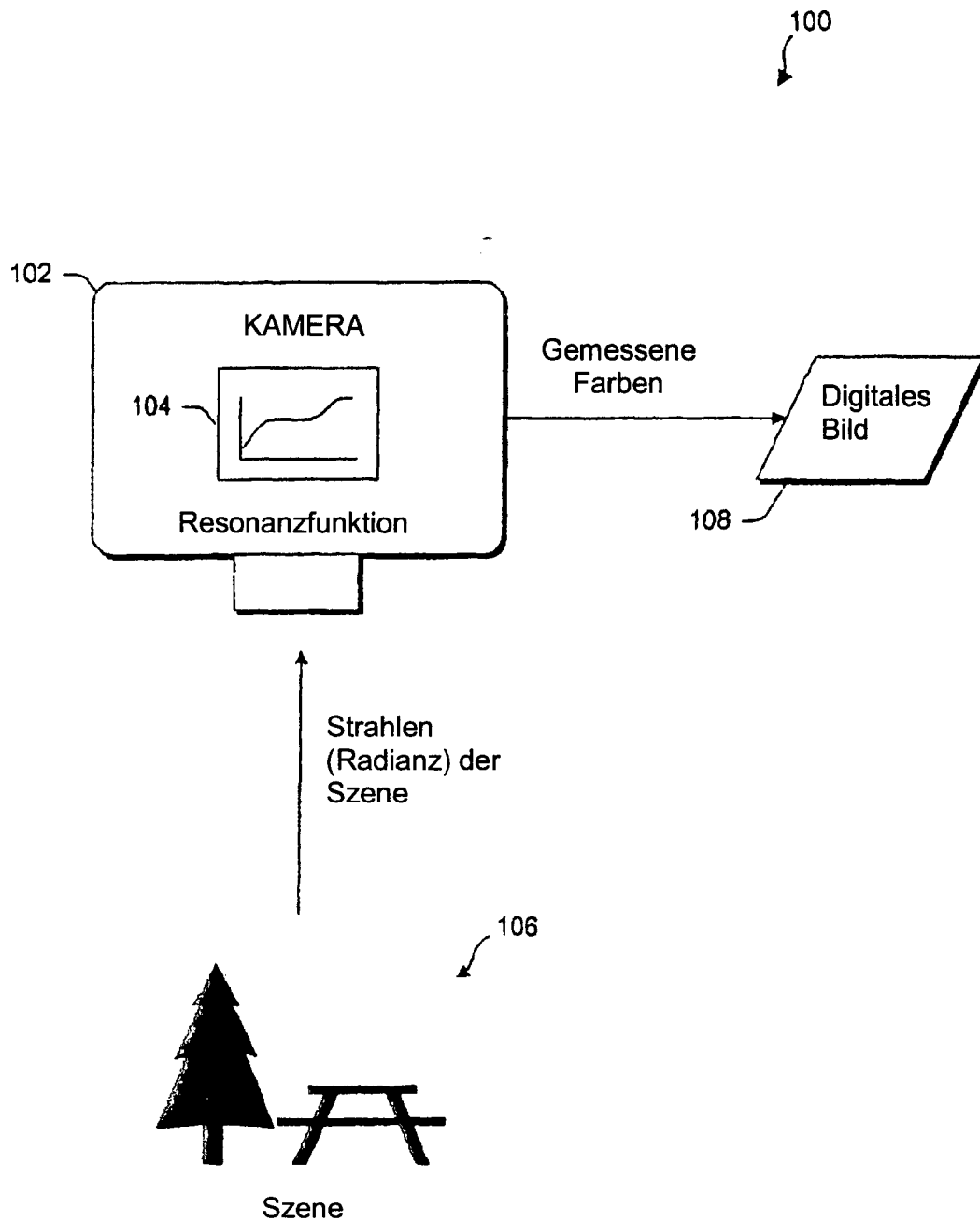


FIG. 1 (Stand der Technik)

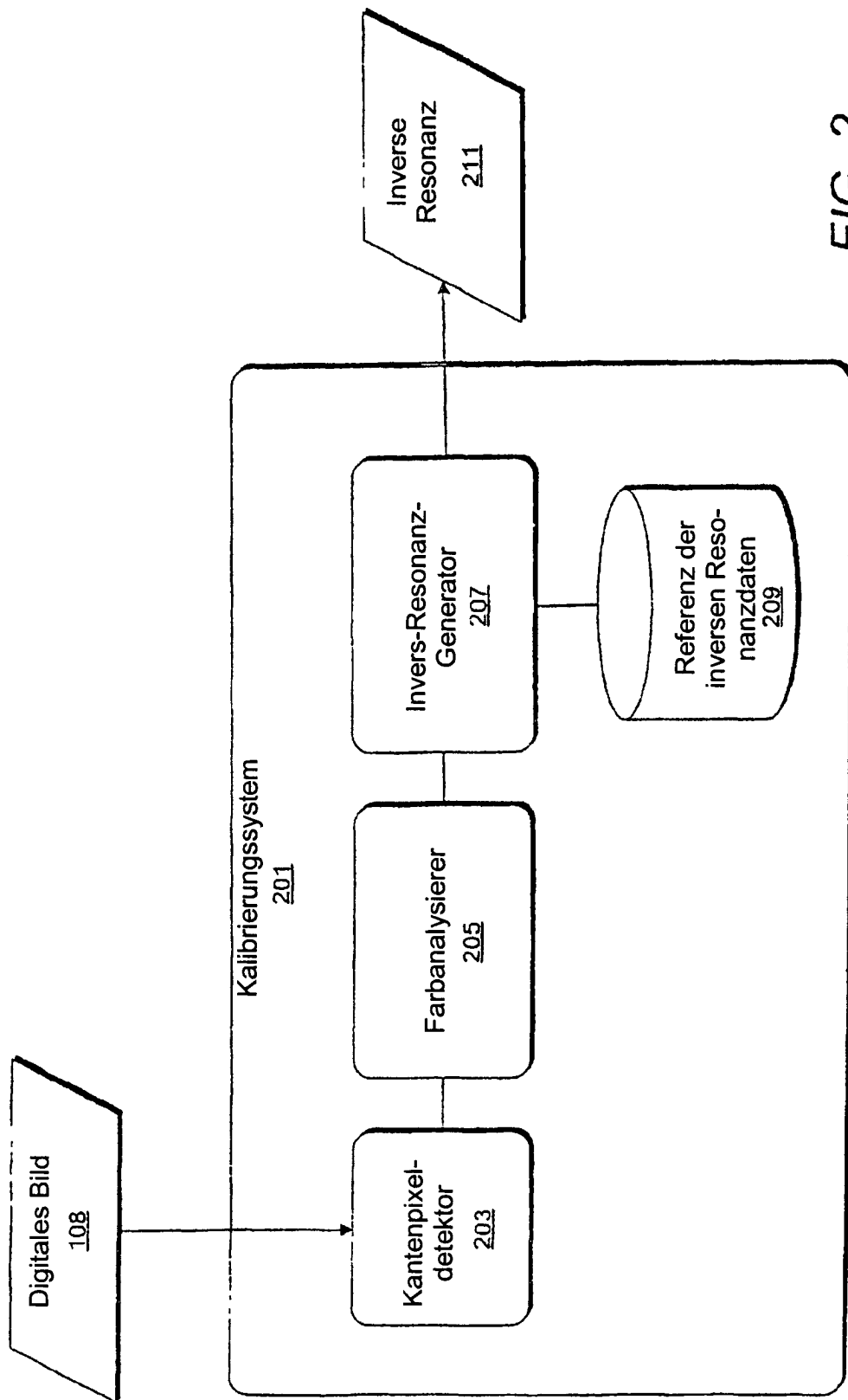


FIG. 2

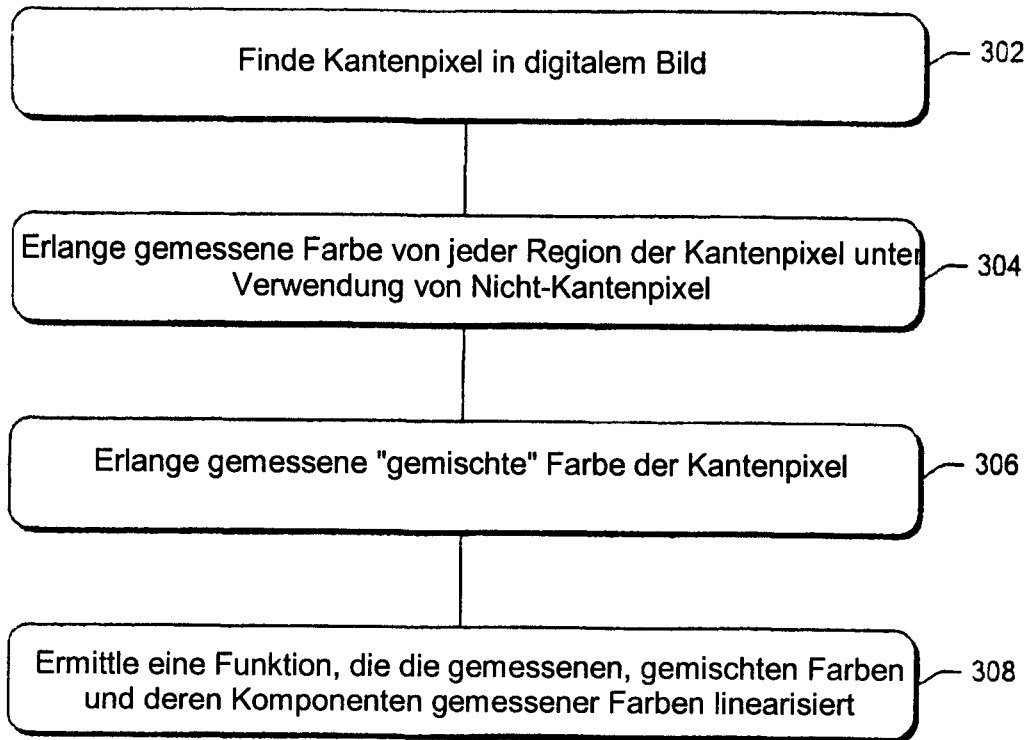


FIG. 3

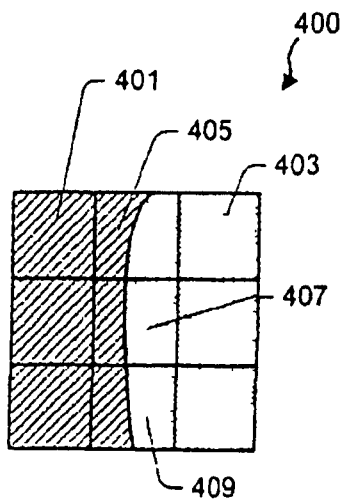


FIG. 4

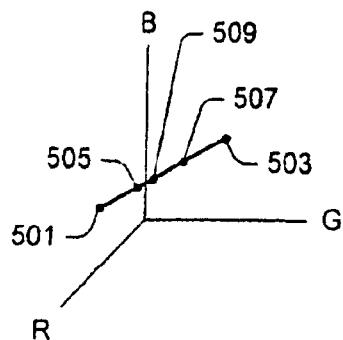


FIG. 5

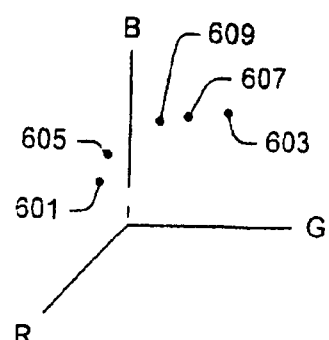


FIG. 6

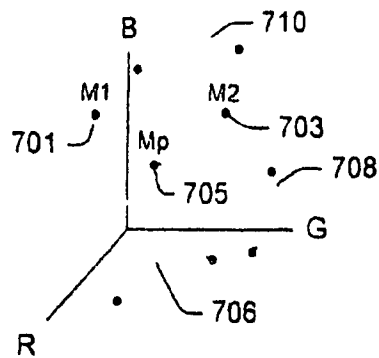


FIG. 7

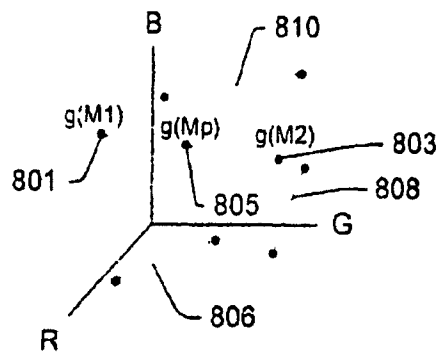


FIG. 8

308

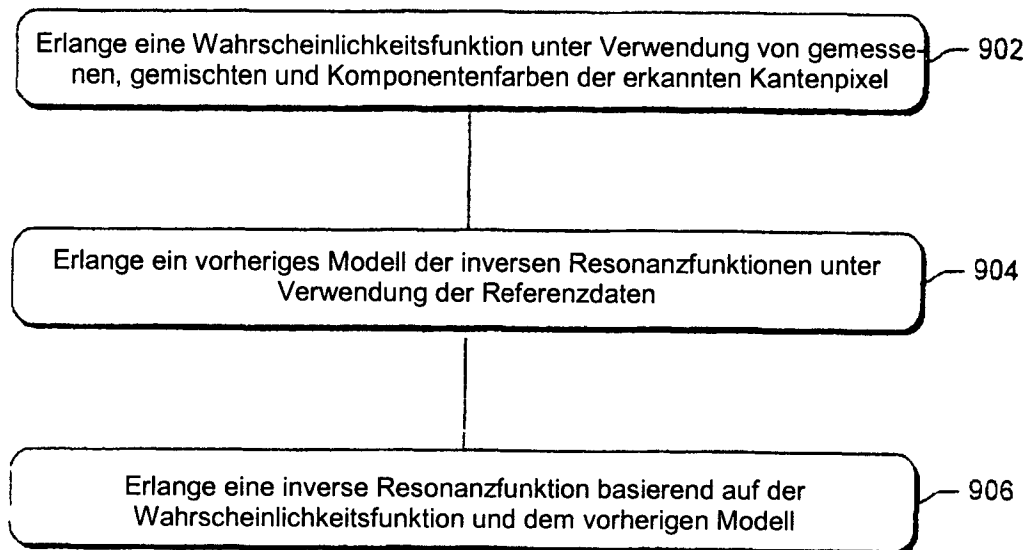


FIG. 9

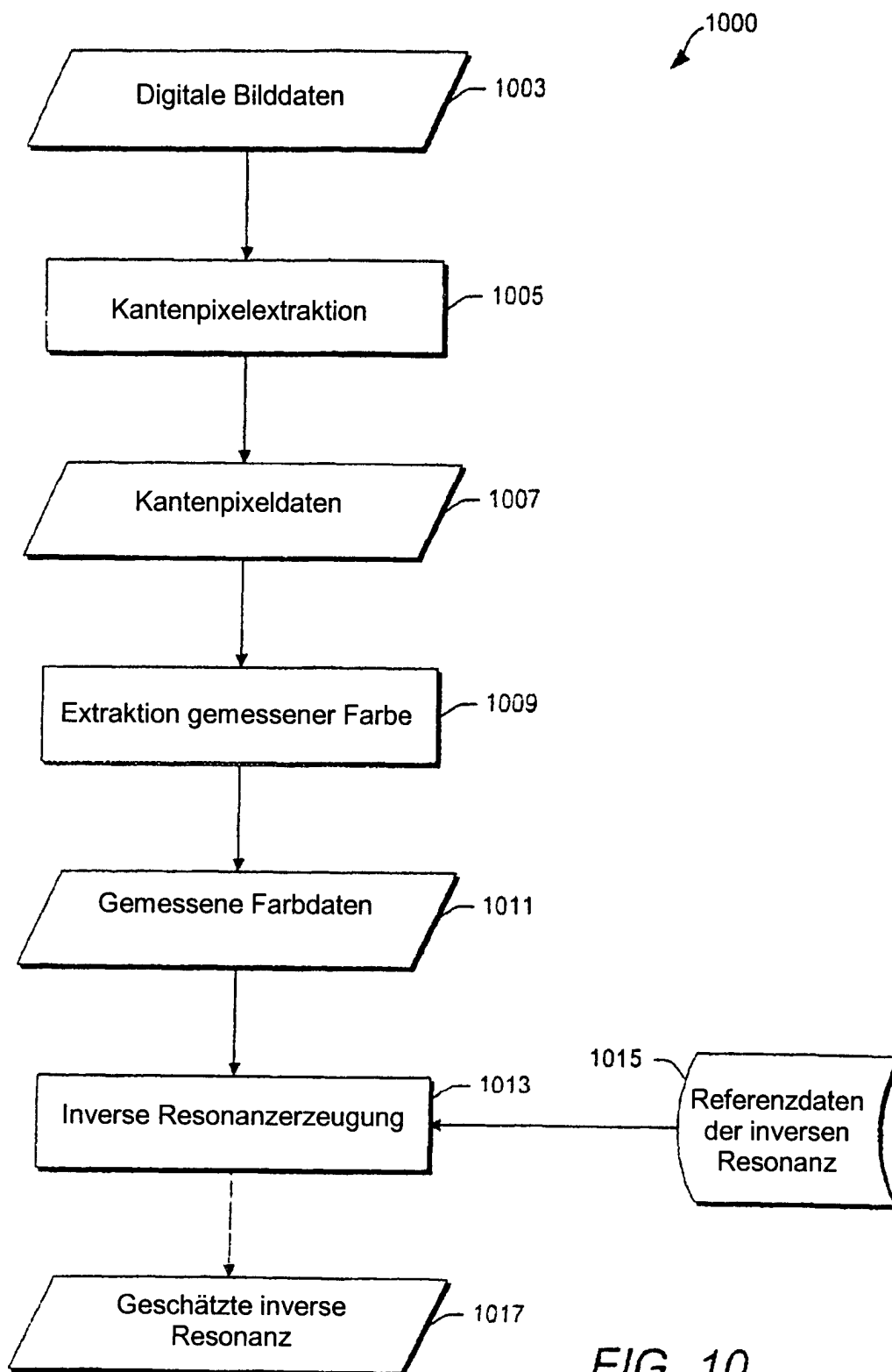


FIG. 10

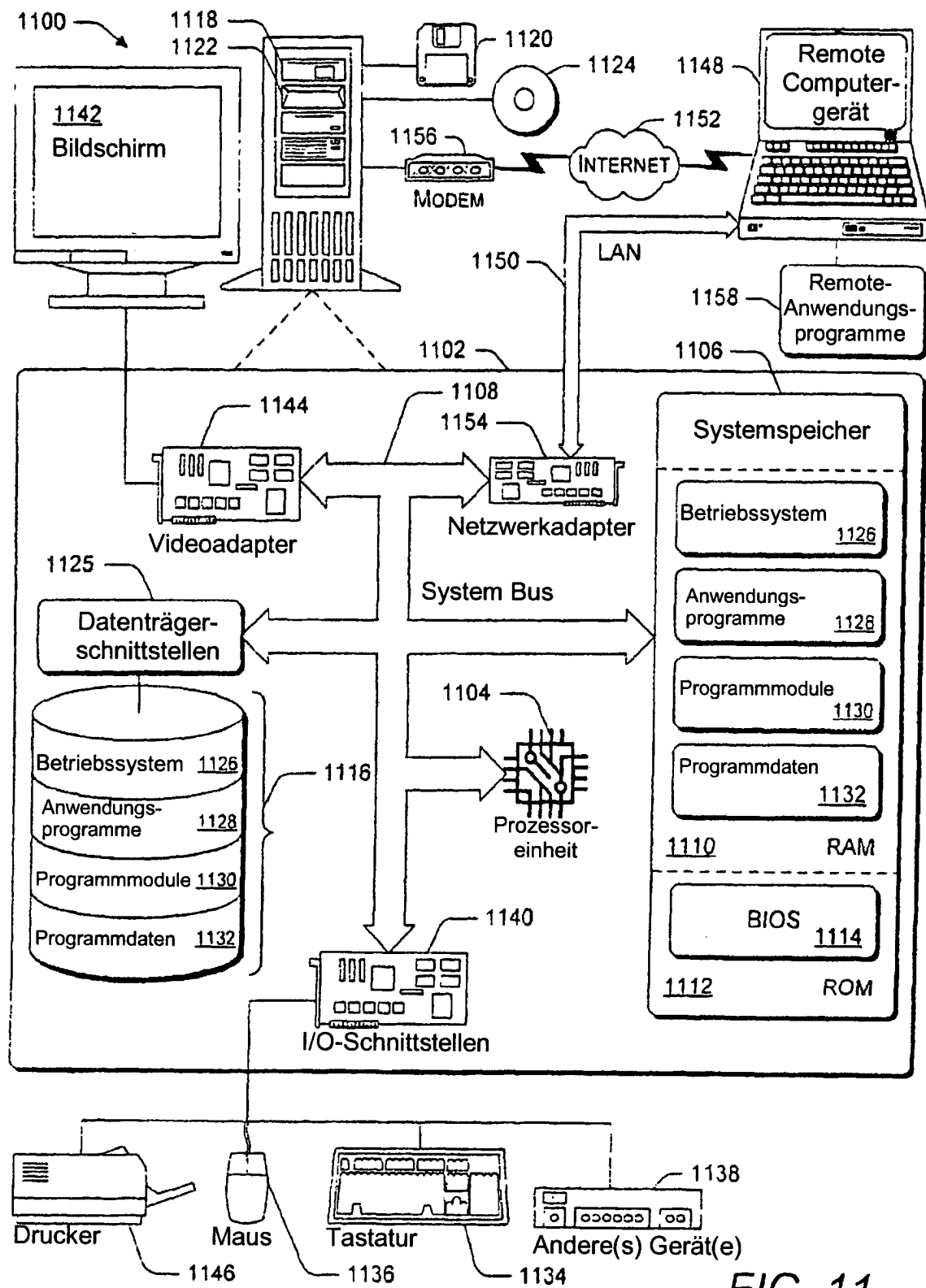


FIG. 11