



(19)
 Bundesrepublik Deutschland
 Deutsches Patent- und Markenamt

(10) **DE 103 11 714 B4** 2008.06.26

(12)

Patentschrift

(21) Aktenzeichen: **103 11 714.8**
 (22) Anmeldetag: **17.03.2003**
 (43) Offenlegungstag: **13.11.2003**
 (45) Veröffentlichungstag
 der Patenterteilung: **26.06.2008**

(51) Int Cl.⁸: **G06T 1/20** (2006.01)
G06F 9/38 (2006.01)
G06F 15/82 (2006.01)
G06F 3/12 (2006.01)

Innerhalb von drei Monaten nach Veröffentlichung der Patenterteilung kann nach § 59 Patentgesetz gegen das Patent Einspruch erhoben werden. Der Einspruch ist schriftlich zu erklären und zu begründen. Innerhalb der Einspruchsfrist ist eine Einspruchsgebühr in Höhe von 200 Euro zu entrichten (§ 6 Patentkostengesetz in Verbindung mit der Anlage zu § 2 Abs. 2 Patentkostengesetz).

(30) Unionspriorität:
2002-076924 19.03.2002 JP

(73) Patentinhaber:
Fuji Xerox Co., Ltd., Tokio/Tokyo, JP

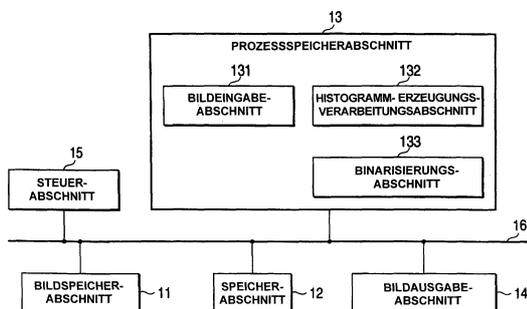
(74) Vertreter:
**Grünecker, Kinkeldey, Stockmair &
 Schwanhäusser, 80802 München**

(72) Erfinder:
**Sekiguchi, Yumi, Ebina, Kanagawa, JP;
 Kumazawa, Yukio, Ebina, Kanagawa, JP; Onda,
 Masanori, Ebina, Kanagawa, JP; Isaka, Youichi,
 Ebina, Kanagawa, JP**

(56) Für die Beurteilung der Patentfähigkeit in Betracht
 gezogene Druckschriften:
US 63 34 181 B1
US 62 52 610 B1

(54) Bezeichnung: **Bildherstellungsvorrichtung und Bildverarbeitungsverfahren**

(57) Hauptanspruch: Bildverarbeitungs-
 vorrichtung, mit einer Mehrzahl von
 Bildverarbeitungsabschnitten (21, 22,
 23), die in einer Pipeline-Art verbun-
 den sind und die jeweilige, vorgegebene
 Prozesse zur Verarbeitung von Bilddaten
 für die Ausgabe auf einem Drucksystem
 durchführen, dadurch gekennzeichnet,
 dass jeder der Bildverarbeitungs-
 abschnitte (21, 22, 23) einen Controller-
 abschnitt (223, 233) zur Übermittlung
 einer Reinitialisierungsinstruktion an
 einen vorhergehenden Bildverarbeitungs-
 abschnitt (21, 22, 23) enthält, wobei
 eine Reinitialisierungsinstruktion den
 vorhergehenden Bildverarbeitungs-
 abschnitt (21, 22, 23) veranlasst,
 die vorgegebene Verarbeitung der
 jeweiligen Bilddaten erneut zu
 beginnen.



Beschreibung

[0001] Die vorliegende Erfindung betrifft eine Bildherstellungsvorrichtung und ein Bildverarbeitungsverfahren, die digitalisierte Bilddaten verarbeiten, und genauer eine Bildherstellungsvorrichtung und ein Bildverarbeitungsverfahren, in denen mehrere komponentenfunktionale Bildverarbeitungsmodulare bereitgestellt werden und verschiedene Bildverarbeitungsfunktionen durch Kombination der Module realisiert werden.

[0002] Eine Technik, bei der vielfache komponentenfunktionale Bildverarbeitungsmodulare bereitgestellt werden und verschiedene Bildverarbeitungsfunktionen durch Kombination der Module realisiert werden, wird in vielen Systemen verwendet, die ein digitales Bild bearbeiten, z. B. in einem Dokumenteneditor, einem Zeichenwerkzeug, einer Bildsendevorrichtung und einem Drucker.

[0003] In einem DTP (Desk Top Publishing) System, in das ein Bild eingegeben werden kann, z. B. in einem Drucksystem, das ein Bild ausgeben kann oder ähnliches, werden verschiedene Bildverarbeitungen wie z. B. eine Vergrößerung/eine Verkleinerung, eine Drehung, eine affine Transformation, eine Farbwandlung, ein Filter und Kombinationen davon mit dem zu verarbeitenden Bild durchgeführt. Wenn ein solcher Prozess durchgeführt werden soll, wird der Prozess manchmal unter Verwendung einer speziellen Hardware in dem Fall durchgeführt, in dem das Attribut eines Eingabebildes, die Prozessinhalte, die Prozedurparameter und ähnliches festgelegt werden. In dem Fall, in dem verschiedene Bilder oder unterschiedliche Farbräume oder Bitanzahlen pro Pixel eingegeben werden oder in dem die Prozessinhalte, die Prozedur, die Parameter oder ähnliches mehrmals geändert werden, müssen jedoch diese Bilder oder Änderungen durch einen flexibleren Aufbau bearbeitet werden.

[0004] Als eine Einrichtung, die dieses Erfordernis erfüllen kann, wurde herkömmlich eine Technik, in der programmierbare Module in einer Pipeline-Art oder einer DAG-Art (DAG = Directed Acyclic Graph = Gelenkter Azyklischer Graph) verbunden sind, derart vorgeschlagen, dass ein gewünschter Prozess flexibel durchgeführt wird (z. B. vergleiche die ungeprüfte japanischen Patentanmeldungen mit den Veröffentlichungsnummern Hei5-260373 und Hei7-105020).

[0005] Die ungeprüfte, japanischen Patentanmeldung mit der Veröffentlichungsnummer Hei5-260373 offenbart eine digitale Videosignalverarbeitungsvorrichtung, die derart aufgebaut ist, dass Inhalte der Berechnungsvorgänge von vielzähligen programmierbaren Berechnungsverarbeitungsabschnitten und die Verbindungsart der programmierbaren Berechnungsverarbeitungsabschnitte durch einen

Netzwerkabschnitt frei von der Außenseite durch einen Hoststeuerabschnitt gesetzt werden können, wodurch ein komplizierter Hochgeschwindigkeitsberechnungsprozess freigegeben wird und der Freiheitsgrad der Änderung einer Funktion oder eines Systems wird erhöht. Die ungeprüfte, japanischen Patentanmeldung mit der Veröffentlichungsnummer Hei7-105020 offenbart ein Pipeline-Bildverarbeitungssystem, in dem notwendige, funktionelle Module in Pipelineart in einer gewünschten Sequenz verbunden/initialisiert sind, um einen Prozess bzw. ein Verfahren durchzuführen, wodurch die Bildverarbeitung flexibel durchgeführt werden kann.

[0006] In der Technik, die in der ungeprüften, japanischen Patentanmeldung mit der Veröffentlichungsnummer Hei7-105020 offenbart ist, wird ein Simulationssystem einer Multiprozess-Pipeline durch einen einzelnen Prozessor, der in einem Betriebssystem, z. B. Unix (registriertes Warenzeichen), verwendet wird, auf ein Bildverarbeitungsfeld oder ähnliches angewendet. Bei der Bildverarbeitung können, insbesondere, wenn eine Prozesseinheit auf einen Teil eines Bildes, z. B. auf eine Linie bzw. Zeile eines Bildes, beschränkt ist, Speicherbereiche, die jeweils für die Verarbeitung durch Verarbeitungsmodulare gehalten werden können, bemerkenswert reduziert werden.

[0007] Im Ergebnis kann eine Bildverarbeitungsvorrichtung, die einen komplexen Prozess mit einer reduzierten Speicherkapazität realisieren kann und die niedrige Herstellungskosten hat, bereitgestellt werden. Wenn eine solche Vorrichtung in einem Betriebssystem, das einen virtuellen Speicher unterstützt, arbeitet, kann ein Auslagern aufgrund einer unzureichenden Speicherkapazität auf einen Minimalwert unterdrückt werden und deshalb kann die Vorrichtung einen Prozess mit einer hohen Geschwindigkeit ausführen.

[0008] In dem Fall, in dem eine Bildverarbeitungsvorrichtung, die einen gewünschten, komplexen Prozess, wie zuvor beschrieben wurde, durchführen kann, durch frei kombinierbare Komponenten-Funktionsmodule aufgebaut ist, ist eine sehr wichtige Funktion die Neuinitialisierung bzw. Wiederinitialisierung, bei der der Zustand in den Anfangszustand derart zurückkehrt, dass ein vorgegebener Prozess wieder mit dem Start von dem Beginn der zu verarbeitenden Bilddaten aus durchgeführt wird. Viele Pipeline-Verarbeitungen können vom Anfang bis zum Ende ohne die Aktivierung der Neuinitialisierung durchgeführt werden. Die Neuinitialisierung muss jedoch in einem Pipeline-Prozess, wie er unten stehend beschrieben wird, aktiviert werden.

[0009] Wie in [Fig. 8](#) gezeigt ist, enthält z. B. eine Pipeline-Verarbeitung einen Bildeingabeabschnitt **102**, einen Histogramm-Erzeugungsabschnitt **103** und einen Binarisierungsabschnitt **104**. Digitale Bild-

daten, die von einem Bildspeicherabschnitt **101** zugeführt werden, werden binarisiert bzw. in eine Binärzahl umgesetzt und die binarisierten Daten werden einem Bildausgabeabschnitt zugeführt. Bei der Pipeline-Verarbeitung wird angenommen, dass Histogramm-Daten eines zu verarbeitenden Bildes durch den Histogramm-Erzeugungsabschnitt **103** erzeugt werden, dass eine Binarisierungsschwelle unter Verwendung dieser Histogramm-Daten bestimmt wird und dass der Binarisierungsabschnitt **104** einen Binarisierungsprozess auf der Basis der binarisierten Schwelle durchführt.

[0010] Da alle Pixelwerte des zu verarbeitenden Bildes notwendig sind, um ein Histogramm in dem Histogramm-Erzeugungsabschnitt **103** erzeugen zu können, ist es erforderlich, Bilddaten aller Zeilen zu lesen und Informationen aller Pixel zu erhalten. Bei der zuvor beschriebenen Pipeline-Verarbeitung der herkömmlichen Art kann eines der nachfolgenden beiden Systeme verwendet werden, um eine Verarbeitung für das Lesen von Bilddaten aller Linien bzw. Zeilen und zum Erhalten von Informationen aller Pixel ausführen zu können.

[0011] In dem ersten der Systeme, wie in [Fig. 9](#) gezeigt ist, gibt es separate Pipelines oder eine Pipeline **200A**, die Histogramm-Daten erzeugt, und eine Pipeline **200B**, die einen Binarisierungsprozess auf der Basis der erzeugten Histogramm-Daten durchführt. Unter der Annahme eines Falles, bei dem dieses System als Prozessverarbeitungsprogramm-Modul auf einem Computer realisiert ist, wird der Bildeingabeabschnitt **102** zweifach erzeugt und der Initialisierungsprozess in dem Bildeingabeabschnitt **102** wird zweimal durchgeführt. Deshalb wird die Quelle und die Verarbeitungsgeschwindigkeit verschwendet.

[0012] In dem zweiten System ist der Histogramm-Erzeugungsabschnitt **103** in der einzelnen Pipeline, die in [Fig. 8](#) gezeigt ist, aufgebaut, wie in [Fig. 10](#) gezeigt ist. Genauer ist der Histogramm-Erzeugungsabschnitt **103** derart aufgebaut, dass er einen Eingangspuffer **301**, der alle Zeilen der Bilddaten, die von dem vorhergehenden Modul aus zugeführt werden, festhält, einen Ausgabepuffer **302**, der ein Verarbeitungsergebnis zu einem Modul in einer nachfolgenden Stufe ausgibt und einen Histogramm-Erzeugungscontroller **303** hat, der den Histogramm-Erzeugungsvorgang steuert.

[0013] Unter Annahme eines Falls, bei dem dieses System als ein Verarbeitungsprogramm-Modul auf einem Computer realisiert ist, werden in dem Histogramm-Erzeugungsabschnitt **103** alle Bilddaten, die von dem vorhergehenden Modul aus bei der Initialisierung zugeführt werden, zuerst in dem Eingabepuffer **301** gespeichert. Als nächstes erzeugt der Histogramm-Erzeugungscontroller **303** Histogramm-Daten unter Verwendung aller Bilddaten, die in dem Eingabe-

bepuffer **301** gespeichert sind, und initialisiert (= 1) einen Zeilenzähler, der intern gespeichert bzw. festgehalten ist. Wenn der Ausgabepuffer **302** von dem nachfolgenden Modul übertragen wird und eine Ausgabeanforderung empfangen wird, kopiert der Histogramm-Erzeugungscontroller **303** die Bilddaten, die in dem Eingabepuffer **301** gehalten sind, in den Ausgabepuffer **302** Zeile für Zeile, gibt den Ausgabepuffer **302** an die nachfolgende Stufe zurück und inkrementiert dann den Zeilenzähler um 1.

[0014] In dem zweiten System ist es erforderlich, obwohl die Wandlung von Bilddaten überhaupt nicht in dem Histogramm-Erzeugungsvorgang des Histogramm-Erzeugungsabschnitts **103** durchgeführt wird, dass alle Eingangsbilddaten gehalten werden. Ein Puffer mit einer Speicherkapazität, durch die alle Bilddaten gespeichert werden können, ist deshalb als der Eingabepuffer **301** notwendig. Im Ergebnis werden Mittel weitgehend verschwendet.

[0015] US-B-6 252 610 beschreibt die Verarbeitung von Grafikobjekten durch eine Folge von sequentiell miteinander verbundenen Verarbeitungsabschnitten. Alle Verarbeitungsabschnitte werden von einem zentralen Prozessor initialisiert und mit Befehlsblöcken für eine entsprechende Verarbeitung versorgt. Jeder Verarbeitungsabschnitt bearbeitet jeweils ein vollständiges Grafikobjekt und gibt das Ergebnis der Verarbeitung an den nachfolgenden Abschnitt weiter.

[0016] US-B-6 334 181 zeigt einen digitalen Signalprozessor. Mehrere Befehle können dabei in einer Pipelinestruktur abgearbeitet werden. Um eine Neuinitialisierung der Pipeline aufgrund einer bedingten Verzweigung in dem Programmablauf zu vermeiden, können einzelne Befehle in der Pipeline durch Leerbefehle (no operation) ersetzt werden.

[0017] Die Erfindung wurde in Hinblick auf die zuvor erwähnten Probleme ausgeführt. Es ist eine Aufgabe der Erfindung, eine Bildverarbeitungsvorrichtung und ein Bildverarbeitungsverfahren bereitzustellen, bei denen, wenn verschiedene Bildverarbeitungsfunktionen durch eine beliebige Kombination von vielfachen komponenten-funktionellen Bildverarbeitungsmodulen realisiert werden, zu verarbeitende Bilddaten wieder gelesen werden können.

[0018] In einem Pipeline-Prozess können, da jeder der Bildverarbeitungsabschnitte (Bildverarbeitungs-module) mit einer Neuinitialisierungsfunktion versehen ist, Bilddaten, die verarbeitet werden sollen, neu gelesen werden. In dem Fall, bei dem ein Vorgang wie z. B. jener, bei dem ein Ergebnis der Analyse von Bilddaten in einem Prozess in einer nachfolgenden Stufe verwendet wird, in der Form eines Programms auf einem Computer realisiert ist, kann deshalb der Prozess, der in der herkömmlichen Realisierung durch mehrmalige Durchgänge durchgeführt worden

ist, durch einen Durchgang durchgeführt werden. Im Ergebnis wird die Programeffizienz erhöht und ein Initialisierungs-Overhead tritt nicht auf, so dass der Prozess vereinfacht werden kann.

[0019] [Fig. 1](#) ist ein Blockdiagramm, das ein Beispiel des Aufbaus einer Bildverarbeitungsvorrichtung zeigt, die eine Ausführungsform der Erfindung ist;

[0020] [Fig. 2](#) ist ein Diagramm, das schematisch den Aufbau von Bilddaten zeigt, die durch die Bildverarbeitungsvorrichtung der Ausführungsform verarbeitet werden können;

[0021] [Fig. 3](#) ist ein Blockdiagramm, das ein Beispiel des Aufbaus einer Bildeingabeeinheit zeigt;

[0022] [Fig. 4](#) ist ein Blockdiagramm, das ein Beispiel für den Aufbau einer Histogramm-Erzeugungsverarbeitungseinheit zeigt;

[0023] [Fig. 5](#) ist ein Blockdiagramm, das ein Beispiel des Aufbaus eines Binarisierungsabschnitts zeigt;

[0024] [Fig. 6](#) ist Flussdiagramm, das den Betrieb der Initialisierung der Bildverarbeitung zeigt;

[0025] [Fig. 7](#) ist ein Flussdiagramm, das den Betrieb einer Ausführung der Bildverarbeitung zeigt;

[0026] [Fig. 8](#) ist ein Blockdiagramm, das ein Beispiel der Kombination von Bildverarbeitungsmodulen zeigt;

[0027] [Fig. 9](#) ist ein Blockdiagramm, das ein Beispiel des Falles zeigt, bei dem der gleiche Vorgang wie der von [Fig. 8](#) ohne Verwendung einer Reinitialisierungsfunktion realisiert ist;

[0028] [Fig. 10](#) ist ein Blockdiagramm, das ein Beispiel der Konfiguration eines Histogramm-Erzeugungsabschnitts in dem Fall zeigt, bei dem der gleiche Prozess wie der von [Fig. 8](#) ohne Verwendung einer Initialisierungsfunktion ausgeführt wird.

[0029] Nachfolgend wird eine Ausführungsform der Erfindung im Detail mit Bezug auf die beiliegenden Zeichnungen beschrieben.

[0030] [Fig. 1](#) ist ein Blockdiagramm, das ein Beispiel des Aufbaus einer Bildverarbeitungsvorrichtung (Systems) zeigt, die eine Ausführungsform der Erfindung ist. Wie aus [Fig. 1](#) ersichtlich ist, hat die Bildverarbeitungsvorrichtung der Ausführungsform der Erfindung einen Bildspeicherabschnitt **11**, einen Speicherabschnitt **12**, einen Verarbeitungsabschnitt **13**, einen Bildausgabeabschnitt **14** und einen Steuerabschnitt **15**. Diese Komponenten sind miteinander über eine Busleitung **16** verbunden. [Fig. 2](#)

zeigt schematisch den Aufbau bzw. die Struktur von Bilddaten, die durch die Bildverarbeitungsvorrichtung verarbeitet werden sollen.

[0031] Gemäß [Fig. 1](#) hält der Bildspeicherabschnitt **11** digitalisierte Bilddaten fest, die verarbeitet werden sollen. Der Speicherabschnitt **12** hält einen Verarbeitungsbereich, der für die Bilddaten erforderlich ist, die in dem Bildspeicherabschnitt **11** gespeichert sind, und verschiedene Verarbeitungen, Zwischenergebnisse von verschiedenen Berechnungen, Prozessparameter usw. Der Bildspeicherabschnitt **11** kann auch als Speicherabschnitt **12** dienen.

[0032] Der Prozessspeicherabschnitt **13** hat verschiedene Bildverarbeitungsabschnitte, die vorgegebene Verarbeitungen der zu verarbeitenden Bilddaten durchführen, die in dem Bildspeicherabschnitt **11** gehalten sind, wie z. B. einen Bildeingabeabschnitt **131**, einen Histogramm-Erzeugungsverarbeitungsabschnitt **132** und einen Binarisierungsabschnitt **133**. Die verschiedenen Bildverarbeitungsabschnitte in dem Prozessspeicherabschnitt **13** können derart arbeiten, dass sie als Programmmodule in dem Speicherabschnitt **12** aufgestellt sind, oder können teilweise oder insgesamt durch spezielle Hardware aufgebaut sein. Die verschiedenen Bildverarbeitungsabschnitte in dem Prozessspeicherabschnitt **13** werden später im Detail beschrieben.

[0033] Der Bildausgabeabschnitt **14** gibt die Verarbeitungsergebnisse aus, die durch die Verarbeitung von Bilddaten erhalten werden, die in dem Bildspeicherabschnitt **11** durch die verschiedenen Bildverarbeitungsabschnitte in dem Prozessspeicherabschnitt **13** gehalten sind, über die Busleitung **16** zu der Außenseite des gezeigten Systems aus. Der Steuerabschnitt **15** ist durch eine CPU und ähnliches aufgebaut und steuert die Prozesse in dem Bildspeicherabschnitt **11**, dem Speicherabschnitt **12**, dem Prozessspeicherabschnitt **13** und dem Bildausgabeabschnitt **14**.

[0034] Als nächstes werden die verschiedenen Bildverarbeitungsabschnitte, die den Bildspeicherabschnitt **13** bilden, durch Beispielgebung verschiedener Verarbeitungen erläutert.

[0035] Wie vorstehend beschrieben wurde, können die verschiedenen Bildverarbeitungsabschnitte als Prozessprogramm-Module auf einem Computer ausgeführt sein oder können teilweise oder insgesamt durch spezielle Hardware aufgebaut sein. Nachfolgend wird, um die Beschreibung vereinfachen zu können, der Fall als Beispiel beschrieben, bei dem die Abschnitte als Prozessprogramm-Module auf einem Computer realisiert sind.

[0036] [Fig. 3](#) ist ein Blockdiagramm, das ein Beispiel des Aufbaus einer Bildeingabeeinheit **21** zeigt,

in die Bilddaten, die verarbeitet werden sollen, eingegeben werden. Die Bildeingabeeinheit **21** entspricht dem Bildeingabeabschnitt **131** in [Fig. 1](#).

[0037] Die Bildeingabeeinheit **21** hat: einen Ausgabepuffer **211**, der zum Lesen von Bilddaten verwendet wird und der die Bilddaten an einen nachfolgenden Modul ausgibt; einen Bildeingabecontroller **212**, der einen Prozess des Eingebens der Bilddaten steuert; einen Zeilenzähler **213**, der die Anzahl der Zeilen zählt. In diesem Beispiel hat die Bildeingabeeinheit **121** den Zeilenzähler **213**. Wenn ein solcher Zähler nicht notwendig ist, kann der Zeilenzähler **213** weggelassen werden.

[0038] Gemäß [Fig. 1](#) bestimmt der Steuerabschnitt **15** in einem Prozess zum Erzeugen eines Moduls Bilddaten, die in dem Bildspeicherabschnitt **11** gehalten sind, wobei notwendige Informationen und ähnliches aus dem Prozessspeicherabschnitt **13** geholt werden und die Bildeingabeeinheit **21** wird als Programmmodul in dem Speicherabschnitt **12** erzeugt. Bei der Erzeugung werden Bildattributinformationen wie z. B. die Größe der Bilddaten, die Bitanzahl, und der Farbraum erhalten, wird die erhaltene Bildattributinformation in dem Speicherabschnitt **12** gespeichert und schreitet, wie in [Fig. 2](#) gezeigt ist, ein Lesestartpositionszeiger zu der Bilddatenstartposition fort, wo Pixeldaten gespeichert sind, und der Zeilenzähler **213** wird initialisiert.

[0039] Wenn der Ausgabepuffer **211** von dem nachfolgenden Modul übertragen wird und eine Ausgabeanforderung empfangen wird, liest der Bildeingabecontroller **202** sequentiell Pixeldaten entsprechend der Größe des Ausgabepuffers aus, wobei bei der Adresse gestartet wird, die durch den Lesestartpositionszeiger angegeben wird, speichert er die Pixeldaten in dem Ausgabepuffer **211** und inkrementiert er dann den Zeilenzähler **213** um 1.

[0040] Der Bildeingabecontroller **212** führt eine Steuerung auch in dem Fall aus, in dem der Controller von einem nachfolgenden bzw. aufeinanderfolgenden Modul Instruktionen zum Reinitialisieren für die Rückkehr in den Zustand des Prozesses der Bildeingabeeinheit **21** in den Anfangszustand derart empfängt, dass der Prozess wieder mit dem Start von dem Beginn der Bilddaten, die verarbeitet werden sollen, durchgeführt wird. Wenn der Bildeingabecontroller **212** Reinitialisierungsbefehle von dem nachfolgenden Modul empfängt, bewegt nämlich der Controller den Lesestartpositionszeiger zu der Bilddatenstartposition und initialisiert den Zeilenzähler **213**.

[0041] [Fig. 4](#) ist ein Blockdiagramm, das ein Beispiel des Aufbaus einer Histogramm-Erzeugungsverarbeitungseinheit **22** zeigt, die einen Prozess des Erzeugens eines Histogramms der Bilddaten durchführt. Die Histogramm-Erzeugungsverarbeitungsein-

heit **22** entspricht dem Histogramm-Erzeugungsverarbeitungsabschnitt **132** in [Fig. 1](#).

[0042] Die Histogramm-Erzeugungsverarbeitungseinheit **22** hat: einen Eingangspuffer **221**, der temporär Bilddaten speichert, die von einem vorhergehenden Modul aus zugeführt werden; einen Ausgangspuffer **222**, der die Bilddaten zu einem nachfolgenden Modul ausgibt; einen Histogramm-Erzeugungscontroller **223**, der einen Histogramm-Erzeugungsprozess steuert; und einen Zeilenzähler **224**, der die Anzahl der Zeilen zählt.

[0043] Gemäß [Fig. 1](#) bestimmt der Steuerabschnitt **15** in dem Erzeugungsvorgang eines Moduls eine Adresse, wo Histogrammdaten in dem Speicherabschnitt gespeichert werden sollen, wobei notwendige Informationen und ähnliches aus dem Prozessspeicherabschnitt **13** geholt werden und die Histogramm-Erzeugungsverarbeitungseinheit **22** als Programm-Modul in dem Speicherabschnitt **12** erzeugt wird. Bei der Erzeugung wird der Histogramm-Erzeugungscontroller **223** mit Informationen über die Verbindung zu dem vorhergehenden Modul von dem Steuerabschnitt **15** versehen, gibt er eine Anforderung zum Erhalten von Attributinformationen von den Eingangsbilddaten zu dem vorhergehenden Modul auf der Basis der Informationen aus und erhält er Bildattributinformationen wie z. B. die Größe der Bilddaten und die Bitanzahl. Der Eingangspuffer **221**, der für einen Prozess notwendig ist, wird in dem Speicherabschnitt **12** ausgebildet und der Zeilenzähler **224** wird initialisiert.

[0044] Die Histogramm-Erzeugungsverarbeitungseinheit **22** wiederholt weiterhin den nachfolgenden Vorgang, bis der Wert des Zeilenzählers **224** gleich der Höhe bzw. dem Wert (Gesamtzeilenanzahl) der eingegebenen Bilddaten ist, wodurch ein Histogramm erzeugt wird. Bei dem Prozess wird der Eingangspuffer **221** zu dem vorhergehenden Modul übertragen, um die Ausgabe anzufordern, wobei Histogrammdaten auf der Basis der Werte der Bilddaten, die in dem Eingangspuffer **221** gespeichert sind, der von der vorhergehenden Stufe zurück gegeben wird, gezählt werden und der Zeilenzähler **224** wird um 1 inkrementiert.

[0045] Wenn die Erzeugung der Histogrammdaten beendet ist, gibt der Histogramm-Erzeugungscontroller **223** den Eingangspuffer **221** frei und initialisiert den Zeilenzähler **224**.

[0046] Wenn der Ausgabepuffer **222** von dem nachfolgenden Modul übertragen wird und eine Ausgabeanforderung zum erstmalig Empfangen wird, gibt der Histogramm-Erzeugungscontroller **223** die Instruktionen für die Reinitialisierung des vorhergehenden Moduls aus, transferiert er dann den Ausgabepuffer **222** zu dem vorhergehenden Modul, um

eine Ausgangsanforderung auszugeben, und gibt er dann das Ergebnis der Anforderung, wie es ist, zu dem nachfolgenden Modul zurück. Wenn die Ausgabeanforderung von dem nachfolgenden Modul für ein zweites Mal und die nachfolgenden Male empfangen wird, transferiert der Histogramm-Erzeugungscontroller den Ausgangspuffer **222** zu dem vorhergehenden Modul und gibt eine Ausgabeanforderung aus und gibt ein Ergebnis der Anforderung, wie es ist, zu dem nachfolgenden Modul zurück.

[0047] In der so aufgebauten Histogramm-Erzeugungsverarbeitungseinheit **22** gibt, wenn eine Ausgabeanforderung zum ersten Mal von dem nachfolgenden Modul empfangen wird, der Histogramm-Erzeugungscontroller **223** Reinitialisierungsinstruktionen zu dem vorhergehenden Modul aus. In Alternative können unabhängig davon, ob eine Ausgabeanforderung von dem nachfolgenden Modul empfangen wird oder nicht, Reinitialisierungsinstruktionen zu dem vorhergehenden Modul gegeben werden, nachdem die Histogramm Daten in dem Modulerzeugungsprozess erzeugt worden sind.

[0048] Wenn der Histogramm-Erzeugungscontroller **223** Reinitialisierungsinstruktionen von dem nachfolgenden Modul empfängt, steuert der Controller auch den Prozess des Sendens von Reinitialisierungsinstruktionen zu dem vorhergehenden Modul. Genauer gibt der Histogramm-Erzeugungscontroller **223**, wenn Reinitialisierungsinstruktionen zu der Histogramm-Erzeugungsverarbeitungseinheit **22** von dem nachfolgenden Modul gegeben werden, Reinitialisierungsinstruktionen zu dem vorhergehenden Modul aus.

[0049] [Fig. 5](#) ist ein Blockdiagramm, das ein Beispiel des Aufbaus eines Binarisierungsabschnitts **23** zeigt, der eine Binarisierungsschwelle unter Verwendung der Histogramm Daten, die durch die Histogramm-Erzeugungsverarbeitungseinheit **22**, die in [Fig. 4](#) gezeigt ist, erzeugt worden sind, bestimmt und der einen Binarisierungsprozess durchführt. Der Binarisierungsabschnitt **23** entspricht dem Binarisierungsabschnitt **133** in [Fig. 1](#).

[0050] Der Binarisierungsabschnitts **23** hat: einen Eingangspuffer **231**, der Bilddaten, die von dem vorhergehenden Modul zugeführt worden sind, speichert; einen Ausgangspuffer **232**, der für das Ausgeben eines Prozessergebnisses zu dem nachfolgenden Modul verwendet wird; und einen Binarisierungscontroller **233**, der einen Binarisierungsprozess steuert.

[0051] Gemäß [Fig. 1](#) empfängt der Binarisierungsabschnitt **23** in einem Prozess zum Erzeugen eines Moduls, um eine Binarisierungsschwelle erzeugen zu können, von dem Steuerabschnitt **15** die Histogramm Daten, die durch den Histogramm-Erzeugungsprozessabschnitt **132** (die Histogramm-Erzeugungsprozesseseinheit **22**) erzeugt worden sind, wobei Informationen und ähnliches zum Verarbeiten der Daten aus dem Prozessspeicherabschnitt **13** ausgelesen werden und der Binarisierungsabschnitts **23** als ein Programmmodul in dem Speicherabschnitt **12** erzeugt wird.

[0052] Bei der Erzeugung wird der Binarisierungscontroller **233** mit Informationen der Verbindung zu dem vorhergehenden Modul von dem Steuerabschnitt **15** versorgt, gibt er eine Anforderung zum Erhalten von Attributinformationen der eingegebenen Bilddaten zu dem vorhergehenden Modul auf der Basis der gelieferten Informationen aus, erhält er Bilddattributionen wie z. B. die Größe der Bilddaten und die Bitanzahl und erzeugt er einen notwendigen Eingangspuffer **702** in dem Speicherabschnitt **112**. Die Binarisierungsschwelle wird auf der Basis der Histogramm Daten durch ein vorgegebenes Verfahren bestimmt. Irgendein Verfahren kann als das Verfahren zum Bestimmen der Binarisierungsschwelle auf der Basis der Histogramm-Daten ausgewählt werden.

[0053] Wenn der Ausgangspuffer **232** von dem nachfolgenden Modul übertragen wird und eine Ausgabeanforderung empfangen wird, überträgt der Binarisierungscontroller **233** den Eingangspuffer **231** zu dem vorhergehenden Modul und gibt eine Ausgabeanforderung aus, führt einen Binarisierungsprozess der Bilddaten, die in dem Eingangspuffer **231** gespeichert sind, unter Verwendung der Binarisierungsschwelle durch, speichert ein Ergebnis des Prozesses in den Ausgangspuffer **232** und gibt den Ausgangspuffer zu dem nachfolgenden Modul zurück.

[0054] Der Binarisierungscontroller **233** führt eine Steuerung auch in dem Fall aus, in dem der Controller Initialisierungsinstruktionen von dem nachfolgenden Modul empfängt, und in dem Fall, in dem die Initialisierungsinstruktionen zu dem vorhergehenden Modul gesendet werden sollen. Wenn er Reinitialisierungsinstruktionen von dem nachfolgenden Modul empfangen hat, gibt der Binarisierungscontroller **233** nämlich Reinitialisierungsinstruktionen zu dem vorhergehenden Modul aus.

[0055] In dem Vorherstehenden wird jeder der verschiedenen Bildverarbeitungsabschnitte (Bildverarbeitungsmodule), d. h., die Bildeingabeeinheit **21**, die Histogramm-Erzeugungsverarbeitungseinheit **22** und der Binarisierungsabschnitt **23** erläutert.

[0056] Nachfolgend wird der Betrieb in dem Fall im Detail mit Bezug auf die Flussdiagramme der [Fig. 6](#) und [Fig. 7](#) beschrieben, in dem der Prozess des Blockdiagramms von [Fig. 8](#) oder der Pipeline-Prozess, in dem die Bildverarbeitungsmodule der Bildeingabe der Histogramm-Erzeugungsprozess

und der Binarisierungsprozess nach Art einer Pipeline verbunden sind, in der Bildverarbeitungsvorrichtung, die in [Fig. 1](#) gezeigt ist, implementiert sind. [Fig. 6](#) ist ein Flussdiagramm der Initialisierung der Bildverarbeitung und [Fig. 7](#) ist ein Flussdiagramm der Implementation der Bildverarbeitung nach der Bildverarbeitungsinitialisierung.

[0057] Zuerst wird der Betrieb der Initialisierung mit Bezug auf das Flussdiagramm von [Fig. 6](#) beschrieben. Die Bildeingabeeinheit **21** wird zuerst initialisiert (Schritt S11). Genauer werden Attributinformationen (die Bildgröße, die Farbrauminformationen und ähnliches) der Eingabebilddaten, die verarbeitet werden sollen, erhalten, wird der Lesestartpositionszeiger (vergleiche [Fig. 2](#)) zu der Bilddatenstartposition bewegt und wird der Zeilenzähler **213** initialisiert. Wenn die Initialisierung der Bildeingabeeinheit **21** beendet ist, wird der Betrieb der Histogramm-Erzeugungsvorrichtungseinheit **22** gestartet.

[0058] In der Histogramm-Erzeugungsvorrichtungseinheit **22** werden zuerst Attributinformationen der eingegebenen Bilddaten aus den Informationen über die Verbindung zu dem vorhergehenden Modul (der Bildeingabeeinheit **21**) (Schritt S12) erhalten bzw. gewonnen und der Eingangspuffer **221** wird reserviert (Schritt S13). Zu diesem Zeitpunkt wird der Eingangspuffer **221** nur derart angefordert, dass er reserviert wird, so dass eine Zeile der eingegebenen Bilddaten gespeichert werden kann.

[0059] Als nächstes wird der Zeilenzähler **224** initialisiert (Schritt S14). Der Eingangspuffer **221** wird zu dem vorhergehenden Modul (der Bildeingabeeinheit **21**) übertragen und die Ausgabeanforderung wird ausgegeben (Schritt S15). In der Bildeingabeeinheit **21** werden dann Bilddaten in den Ausgangspuffer **211** (= der Eingabepuffer **221**) eingelesen, der von dem nachfolgenden Modul übertragen wird (der Histogramm-Erzeugungsvorrichtungseinheit **22**), der Zeilenzähler **213** wird um 1 inkrementiert und der Puffer wird an den nachfolgenden Modul (Schritt S16) zurückgegeben.

[0060] In der Histogramm-Erzeugungsvorrichtungseinheit **22** wird danach auf die Bilddaten, die in dem Eingabepuffer **221** gespeichert sind, Bezug genommen, werden die Histogramm-Daten gezählt (Schritt S17) und es wird beurteilt, ob der Zählwert des Zeilenzählers **224** die Höhe bzw. den Wert (Gesamtzeilenanzahl) der Bilddaten (Schritt S18) erreicht oder nicht. Wenn der Wert des Zeilenzählers **224** nicht die Höhe der eingegebenen Bilddaten erreicht, wird der Zeilenzähler **224** um 1 inkrementiert (Schritt S19) und die Steuerung kehrt dann zum Schritt S15 zurück, um die Verarbeitungen der Schritte S15 bis S18 zu wiederholen. Wenn der Wert des Zeilenzählers **224** gleich der Höhe der eingegebenen Bilddaten ist, wird der Eingangspuffer **221** freigege-

ben und der Zeilenzähler **224** wird initialisiert (Schritt S20).

[0061] Als nächstes wird der Betrieb des Binarisierungsabschnitts **23** gestartet. In dem Binarisierungsabschnitts **23** wird zuerst die Binarisierungsschwelle auf der Basis der erzeugten Histogrammdaten (Schritt S21) erhalten. Irgendein Verfahren kann als das Verfahren zum Bestimmen dieser Binarisierungsschwelle auf der Basis der Histogrammdaten ausgewählt werden. In dem Binarisierungsabschnitt **23** werden danach Attributinformationen der eingegebenen Bilddaten aus Informationen der Verbindungen zu dem vorhergehenden Modul (der Histogramm-Erzeugungsvorrichtungseinheit **22**) (Schritt S22) erhalten und der Eingangspuffer **231** wird reserviert (Schritt S23). Als Ergebnis der vorstehenden Serien von Verarbeitungen wird die Initialisierung beendet.

[0062] In dem Flussdiagramm, das in [Fig. 6](#) gezeigt ist, werden die Verarbeitungen in dem Binarisierungsabschnitts **23** in der Sequenz der Schritt S21 → Schritt S22 durchgeführt. Wenn Bildattributinformationen zur Bestimmung der Binarisierungsschwelle nicht notwendig sind, kann die Sequenz derart geändert werden, dass die Prozesse in der Sequenz von z. B. Schritt S22 → Schritt S23 → Schritt S21 durchgeführt werden.

[0063] Als nächstes wird der Betrieb der Implementation der Bildverarbeitung nach der Bildverarbeitungsinitialisierung mit Bezug auf das Flussdiagramm von [Fig. 7](#) beschrieben. Nachdem die Bildverarbeitungsinitialisierung beendet worden ist, ist der Zeilenzähler **213** der Bildeingabeeinheit **21** in dem Zustand, in dem der Wert gleich dem Wert bzw. der Höhe der Bilddaten ist.

[0064] Zuerst erhält der Bildeingabeabschnitt **14** zuvor Attributinformationen des Bildes, das ausgegeben werden soll, und reserviert einen Ausgangspuffer, der eine Zeile des auszugebenden Bildes speichern kann (zum Zwecke der Vereinfachung wird der Puffer als Ausgangspuffer X bezeichnet) (Schritt S31). Danach überträgt der Bildausgabeabschnitt **14** den Ausgangspuffer X zu dem vorhergehenden Modul (dem Binarisierungsabschnitts **23**) und gibt eine Ausgabeanforderung (Schritt S32) aus. In Antwort auf die Ausgabeanforderung überträgt der Binarisierungsabschnitts **23** den Eingabepuffer **231** zu dem vorhergehenden Modul (der Histogramm-Erzeugungsvorrichtungseinheit **22**) und gibt eine Ausgabeanforderung (Schritt S33) aus.

[0065] Danach wird beurteilt, ob die Histogramm-Erzeugungsvorrichtungseinheit **22** die Ausgabeanforderung von dem nachfolgenden Modul zum erstenmal empfängt oder nicht (Schritt S34). Wenn die Einheit die Ausgabeanforderung zum ers-

tenmal empfängt, gibt die Histogramm-Erzeugungsverarbeitungseinheit **22** die Reinitialisierungsinstruktionen zu dem vorhergehenden Modul aus (der Bildeingabeeinheit **21**) (Schritt S35). Wenn die Bildeingabeeinheit **21** die Reinitialisierungsinstruktionen von dem nachfolgenden Modul (der Histogramm-Erzeugungsverarbeitungseinheit **22**) empfängt, bewegt die Einheit den Lesestart-Positionierer, der in [Fig. 2](#) gezeigt ist, zu der Bilddatenstartposition und initialisiert den Zeilenzähler **213** (Schritt S36).

[0066] Der Reinitialisierungsprozess in den Schritten S35 und S36 ist ein charakteristischer Abschnitt der Erfindung. In der Ausführungsform der Erfindung wird der Reinitialisierungsprozess in der Mitte der Ausführung der Bildverarbeitung nach der Initialisierung durchgeführt. In Alternative kann der Reinitialisierungsprozess zu dem Zeitpunkt bzw. zu der Zeit ausgeführt werden, wenn der Initialisierungsprozess des Flussdiagramms von [Fig. 6](#) beendet ist.

[0067] Nach dem Reinitialisierungsprozess oder, wenn der Empfang der Ausgabeanforderung von dem nachfolgenden Modul durch die Histogramm-Erzeugungsverarbeitungseinheit **22** nicht das erstemal vorliegt, überträgt die Histogramm-Erzeugungsverarbeitungseinheit **22** den Ausgabepuffer **222**, der von dem nachfolgenden Modul (dem Binarisierungsabschnitts **23**) übertragen wird, zu dem vorhergehenden Modul (der Bildeingabeeinheit **21**) und gibt eine Ausgabeanforderung (Schritt S37) aus. In der Bildeingabeeinheit **21** werden Bilddaten in den Ausgangspuffer **211** (= der Ausgangspuffer **222**) eingelesen, der von dem nachfolgenden Modul (der Histogramm-Erzeugungsverarbeitungseinheit **22**) übertragen wird, wird der Zeilenzähler **213** um 1 inkrementiert und wird dann der Puffer zu der nachfolgenden Stufe (Schritt S38) zurückgegeben. Die Histogramm-Erzeugungsverarbeitungseinheit **22** gibt den Ausgangspuffer **222**, der von dem vorhergehenden Modul zurückgegeben wird (der Bildeingabeeinheit **21**), zu der nachfolgenden Stufe (Schritt S39) zurück.

[0068] Der Binarisierungsabschnitt **23** binarisiert Bilddaten, die in dem Eingangspuffer **231** gespeichert sind (= dem Ausgangspuffer **222**), der von dem vorhergehenden Modul (der Histogramm-Erzeugungsverarbeitungseinheit **22**) zurückgegeben wird, speichert ein Ergebnis des Prozesses in dem Ausgangspuffer **232** (= dem Ausgangspuffer X) und gibt den Puffer zu dem nachfolgenden Modul zurück (dem Bildausgabeabschnitt **14**) (Schritt S40).

[0069] Als nächstes wird beurteilt, ob der Zählwert des Zeilenzählers **213** gleich der Höhe der eingegebenen Bilddaten ist oder nicht (Schritt S41). Wenn der Wert nicht die Höhe der Bilddaten erreicht, wird der Zeilenzähler **213** um 1 inkrementiert (Schritt S41) und die Steuerung kehrt dann zum Schritt S32 zu-

rück, um die Prozesse der Schritte S32 bis S41 wiederholen zu können. Wenn der Wert des Zeilenzählers **213** gleich der Höhe der eingegebenen Bilddaten ist, ist die zuvor beschriebene Serie von Prozessen beendet.

[0070] Wie zuvor beschrieben wurde, ist in einem Pipeline-Prozess, in dem mehrere Bildverarbeitungsabschnitte (Bildverarbeitungsmodule), die jeweilige, vorgegebene Verarbeitungen durchführen, auf Pipeline-Art verbunden sind, jeder der Bildverarbeitungsabschnitte mit der Reinitialisierungsfunktion versehen. Auch in dem Fall, in dem ein Bildattribut, z. B. Histogramm Daten, unter Verwendung aller Pixelwerte der Bilddaten erhalten wird und danach eine nachfolgende Stufe einen Prozess unter Verwendung der erhaltenen Histogramm Daten durchführt, können deshalb gewünschte Bildverarbeitungen durchgeführt werden, ohne dass die gesamten Bilddaten in einem Puffer gehalten werden müssen oder dass ein Initialisierungsprozess mehrmals ausgeführt werden muss.

[0071] In der vorstehend beschriebenen Ausführungsform wird die Erfindung auf eine Pipeline-Verarbeitung angewendet, in der die Bildeingabeeinheit **21**, die Histogramm-Erzeugungsverarbeitungseinheit **22** und der Binarisierungsabschnitt **23** als Bildverarbeitungsabschnitte in Pipeline-Art verbunden sind. Die Erfindung ist jedoch nicht auf dieses Anwendungsbeispiel beschränkt und kann auf alle Pipeline-Verarbeitungen einschließlich Bildverarbeitungsabschnitte (Bildverarbeitungsmodule) angewendet werden, die einen Prozess zum Erhalten eines Bildattributs wie z. B. einer projizierten Verteilung unter Verwendung aller Pixelwerte zu verarbeitender Bilddaten durchführt.

[0072] Wie vorstehend beschrieben wurde, wird gemäß der Erfindung jeder der Bildverarbeitungsabschnitte, wenn mehrere Komponenten funktionelle Bildverarbeitungsmodule frei kombiniert werden, um verschiedene Bildverarbeitungsfunktionen ausführen zu können, mit einer Reinitialisierungsfunktion versehen, wodurch Bitdaten, die verarbeitet werden sollen, neu bzw. wieder gelesen werden können. In dem Fall, in dem ein Prozess, z. B. ein Prozess, in dem ein Ergebnis der Analyse der Bilddaten in einen Prozess einer nachfolgenden Stufe verwendet wird, in der Form eines Programms auf einem Computer ausgeführt wird, kann deshalb der Prozess, der mit mehreren Durchgängen herkömmlich durchgeführt worden ist, nun durch einen Durchgang durchgeführt werden. Im Ergebnis wird die Programmiereffizienz erhöht und der Initialisierungsoverhead tritt auf, so dass der Prozess vereinfacht werden kann.

Bezugszeichenliste

[\[Fig. 6\]](#)[\[Fig. 1\]](#)

| | |
|------------|---|
| 11 | Bildspeicherabschnitt |
| 12 | Speicherabschnitt |
| 13 | Prozessspeicherabschnitt |
| 14 | Bildausgabeabschnitt |
| 15 | Steuerabschnitt |
| 131 | Bildeingabeabschnitt |
| 132 | Histogramm-Erzeugungsverarbeitungsabschnitt |
| 133 | Binarisierungsabschnitt |

[\[Fig. 2\]](#)

| | |
|----------|---|
| a | Initialisierungsposition für Lesestartpositionszeiger |
| b | Bilddatenstartposition |
| c | Bildattributinformation |
| d | Pixeldaten |
| e | Breite bzw. Weite der Bilddaten |
| f | Höhe der Bilddaten |

[\[Fig. 3\]](#)

| | |
|------------|--------------------------------------|
| 21 | Bildeingabeeinheit |
| 211 | Ausgabepuffer |
| 212 | Bildeingabecontroller |
| 213 | Zeilenzähler |
| a | Bildspeicherabschnitt |
| b | Reinitialisierungssteuerung |
| c | Eingabepuffer für nachfolgende Stufe |

[\[Fig. 4\]](#)

| | |
|------------|---|
| 22 | Histogramm-Erzeugungsverarbeitungseinheit |
| 221 | Eingangspuffer |
| 222 | Ausgangspuffer |
| 223 | Histogramm-Erzeugungscontroller |
| 224 | Zeilenzähler |
| a | Ausgangspuffer der vorhergehenden Stufe |
| b | Controller der vorhergehenden Stufe |
| c | Ausgangspuffer der vorhergehenden Stufe |
| d | Reinitialisierungssteuerung |
| e | Controller der nachfolgenden Stufe |
| f | Eingangspuffer der nachfolgenden Stufe |

[\[Fig. 5\]](#)

| | |
|------------|---|
| 23 | Binarisierungsabschnitt |
| 231 | Eingangspuffer |
| 232 | Ausgangspuffer |
| 233 | Binarisierungscontroller |
| a | Ausgangspuffer der vorhergehenden Stufe |
| b | Controller der vorhergehenden Stufe |
| c | Controller der nachfolgenden Stufe |
| d | Eingangspuffer der nachfolgenden Stufe |
| e | Reinitialisierungssteuerung |

| | |
|------------|--|
| S11 | Initialisiere Bildeingabeeinheit (Zeilenzähler 213 = 1) |
| S12 | Erhalte Attributinformationen von Modul des vorhergehenden Moduls |
| S13 | Reserviere Eingangspuffer S21 |
| S14 | Zeilenzähler 224 = 1 |
| S15 | Übertrage Eingangspuffer 221 zu vorhergehenden Modul und gebe Ausgabeanforderung aus |
| S16 | Bildeingabeeinheit liest Bilddaten in den Ausgangspuffer 211 , inkrementiert den Zeilenzähler 213 um 1 und gibt Puffer zu nachfolgender Stufe zurück |
| S17 | Beziehe auf Eingangspuffer 221 und zähle Histogramm Daten |
| S18 | Zeilenzähler 224 = Höhe der Bilddaten? |
| S19 | Zeilenzähler 224 + 1 |
| S20 | Freigeben Eingangspuffer 221 , Zeilenzähler 224 = 1 |
| S21 | Bestimme Binarisierungsschwelle unter Verwendung von Histogramm Daten |
| S22 | Erhalte Attributinformationen von vorhergehendem Modul |
| S23 | Reserviere Eingangspuffer 231 |
| a | Initialisiere Histogramm-Erzeugungsverarbeitungsabschnitt |
| b | Start der Initialisierung |
| c | Ende der Initialisierung |

[\[Fig. 7\]](#)

| | |
|------------|--|
| S31 | Reserviere Ausgangspuffer X |
| S32 | Bildausgabeabschnitt überträgt Ausgangspuffer X zu dem Binarisierungsabschnitt und gibt Ausgabeanforderung aus |
| S33 | Binarisierungsabschnitt überträgt Eingangspuffer 231 zu Histogramm-Erzeugungsverarbeitungseinheit und gibt Ausgabeanforderung aus |
| S34 | Histogramm-Erzeugungsverarbeitungseinheit empfängt Ausgabeanforderung zum ersten Mal? |
| S35 | Histogramm-Erzeugungsverarbeitungseinheit gibt Reinitialisierungsinstruktionen zu Bildeingabeeinheit |
| S36 | In Bildeingabeeinheit, Lesestartpositionierungszeiger = Bilddatenstartposition, Zeilenzähler 213 = 1 |
| S37 | Histogramm-Erzeugungsverarbeitungseinheit überträgt Ausgangspuffer 222 , der von der nachfolgenden Stufe übertragen wird, zu der Bildeingabeeinheit und gibt Ausgabeanforderung aus |
| S38 | Bildeingabeeinheit liest Bilddaten in Ausgangspuffer 211 , inkrementiert Zeilenzähler 213 um 1 und gibt Puffer an nachfolgende Stufe zurück |

- S39** Histogramm-Erzeugungsverarbeitungseinheit gibt Ausgangspuffer **222**, der von der vorhergehenden Stufe zurückgegeben wird, zu der nachfolgenden Stufe zurück
- S40** Binarisierungsabschnitt binarisierte Bilddaten des Eingangspuffers **231**, der von der vorhergehenden Stufe zurückgegeben wird, speichert Ergebnis in Ausgangspuffer **232** und gibt Puffer zum Bildausgabeabschnitt zurück
- S41** Zeilenzähler **213** = Höhe der Bilddaten?
- S42** Zeilenzähler **213** + 1
- a** Starten des Prozesses
- b** Ende des Prozesses
- c** Zeilenzähler **213** = Höhe der Bilddaten

[Fig. 8]

- 101** Bildspeicherabschnitt
- 102** Bildeingabeabschnitt
- 103** Histogramm-Erzeugungsabschnitt
- 104** Binarisierungsabschnitt
- 105** Bildausgabeabschnitt

[Fig. 9]

- 101** Bildspeicherabschnitt
- 102** Bildeingabeabschnitt
- 103** Histogramm-Erzeugungsabschnitt
- 104** Binarisierungsabschnitt
- 105** Bildausgabeabschnitt
- 200A** Pipeline für Erzeugung der Histogramm Daten
- 200B** Pipeline zum Durchführen des Binarisierungsprozesses auf der Basis der Histogramm Daten
- a** Ende

[Fig. 10]

- 103** Histogramm-Erzeugungsabschnitt
- 301** Eingangspuffer (alle Zeilen des Bildes)
- 302** Ausgangspuffer (eine Zeile des Bildes)
- 303** Histogramm-Erzeugungscontroller
- a** Ausgangspuffer der vorhergehenden Stufe
- b** Controller der vorhergehenden Stufe
- c** Controller der nachfolgenden Stufe
- d** Eingangspuffer der nachfolgenden Stufe

Patentansprüche

1. Bildverarbeitungsvorrichtung, mit einer Mehrzahl von Bildverarbeitungsabschnitten (**21**, **22**, **23**), die in einer Pipeline-Art verbunden sind und die jeweilige, vorgegebene Prozesse zur Verarbeitung von Bilddaten für die Ausgabe auf einem Drucksystem durchführen, **dadurch gekennzeichnet**, dass jeder der Bildverarbeitungsabschnitte (**21**, **22**, **23**) einen Controllerabschnitt (**223**, **233**) zur Übermittlung einer Reinitialisierungsinstruktion an einen vorhergehenden

den Bildverarbeitungsabschnitt (**21**, **22**, **23**) enthält, wobei eine Reinitialisierungsinstruktion den vorhergehenden Bildverarbeitungsabschnitt (**21**, **22**, **23**) veranlasst, die vorgegebene Verarbeitung der jeweiligen Bilddaten erneut zu beginnen.

2. Die Bildverarbeitungsvorrichtung nach Anspruch 1, worin mindestens einer der Bildverarbeitungsabschnitte (**21**, **22**, **23**) einen vorgegebenen Prozess unter Verwendung aller Pixelwerte der zu verarbeitenden Bilddaten durchführt.

3. Die Bildverarbeitungsvorrichtung nach Anspruch 2, worin der vorgegebene Prozess ein Prozess zum Erhalten eines Bildattributs ist.

4. Die Bildverarbeitungsvorrichtung nach Anspruch 3, worin das Bildattribut ein Histogramm des zu verarbeitenden Bildes ist.

5. Ein Bildverarbeitungsverfahren in einem Pipelineprozess zur Durchführung jeweiliger, vorgegebener Prozesse zur Verarbeitung von Bilddaten für die Ausgabe auf einem Drucksystem durch eine Vielzahl von Bildverarbeitungsmodulen, die in einer Pipeline-Art verbunden sind, gekennzeichnet durch den Schritt Übermitteln einer Reinitialisierungsinstruktion an ein vorhergehendes Bildverarbeitungsmodul, wobei eine Reinitialisierungsinstruktion das vorhergehende Bildverarbeitungsmodul veranlasst, die vorgegebene Verarbeitung der jeweiligen Bilddaten erneut zu beginnen.

Es folgen 8 Blatt Zeichnungen

FIG. 1

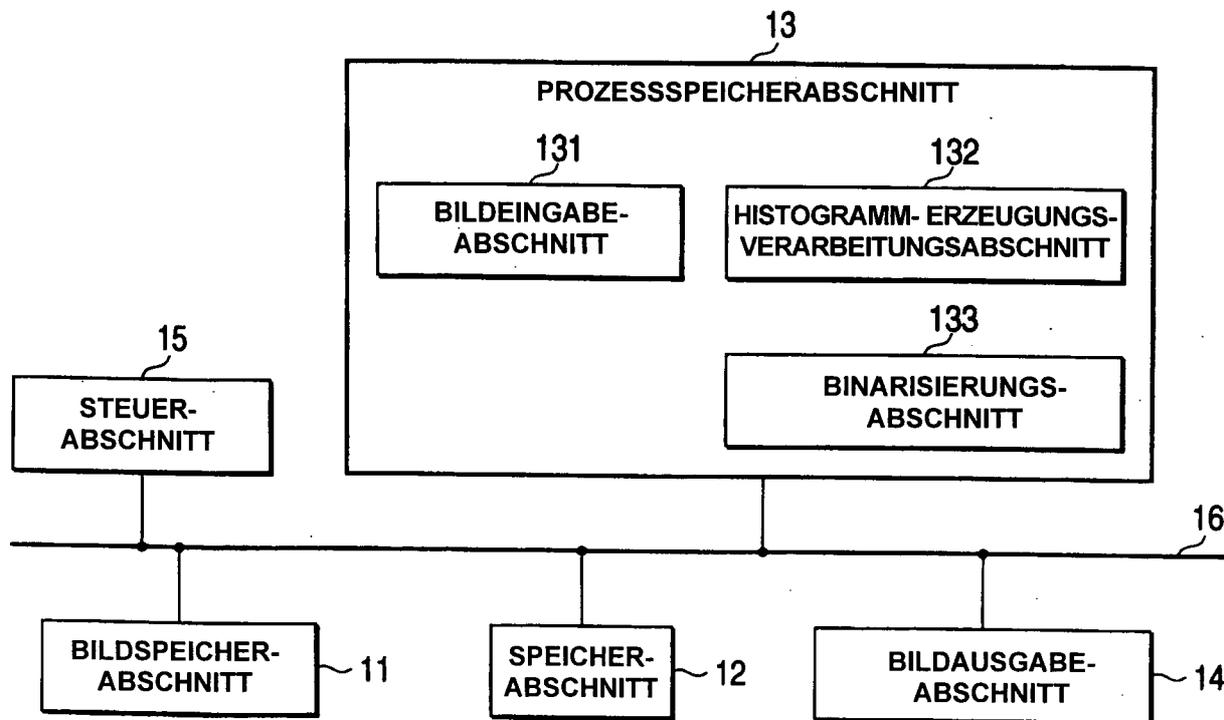


FIG. 2

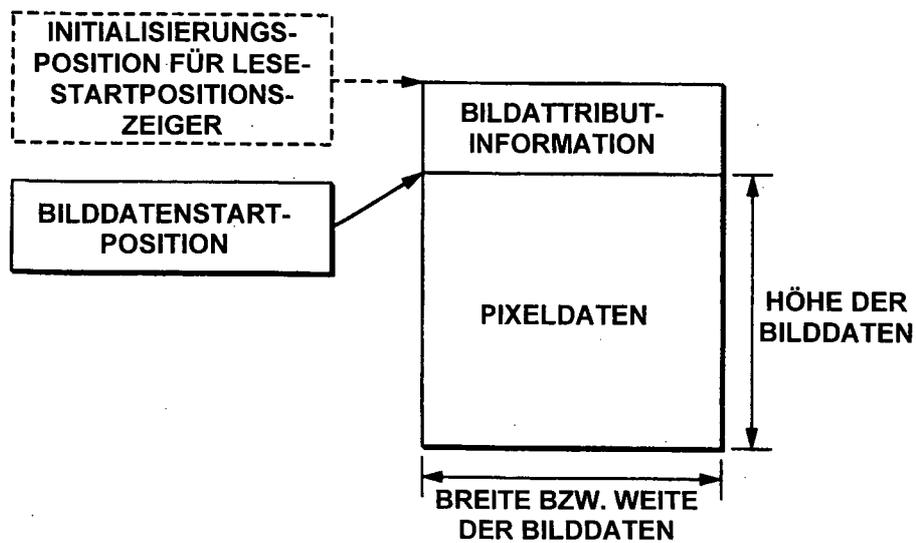


FIG. 3

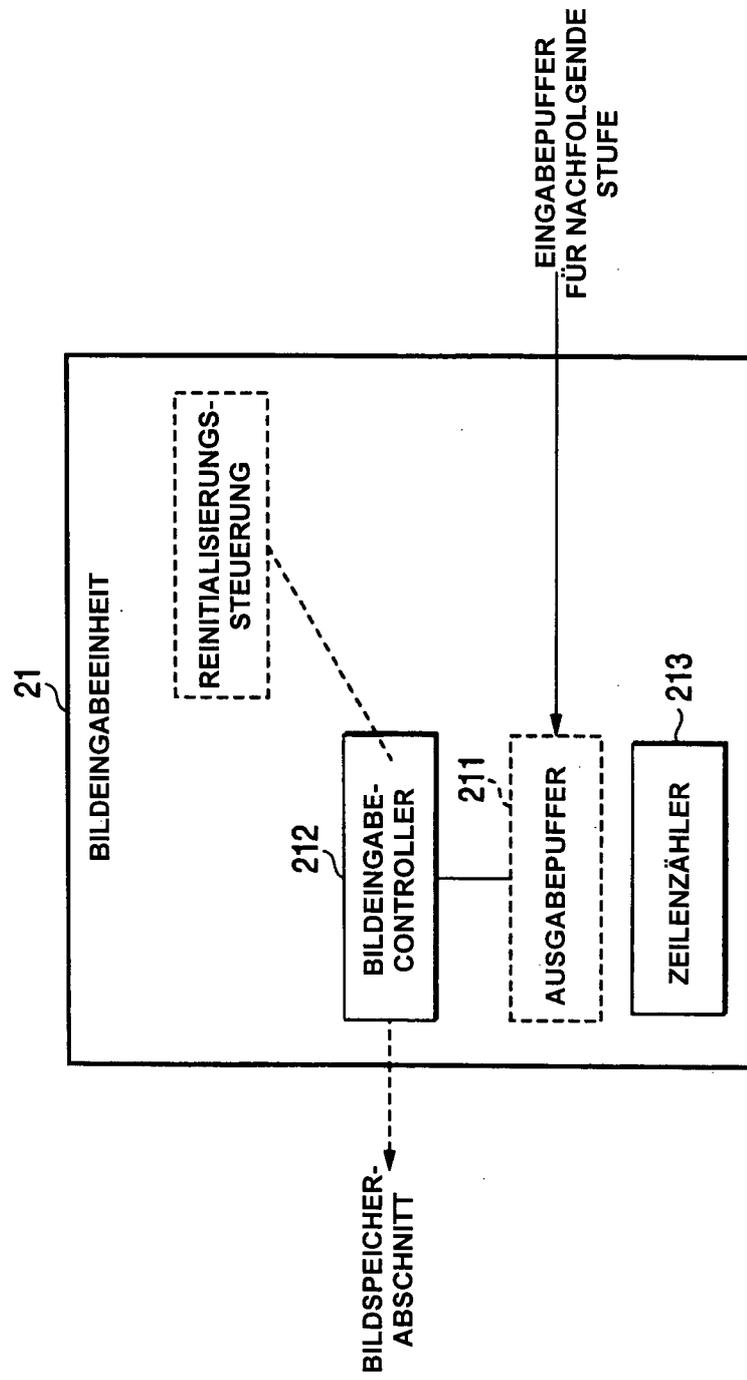


FIG. 4

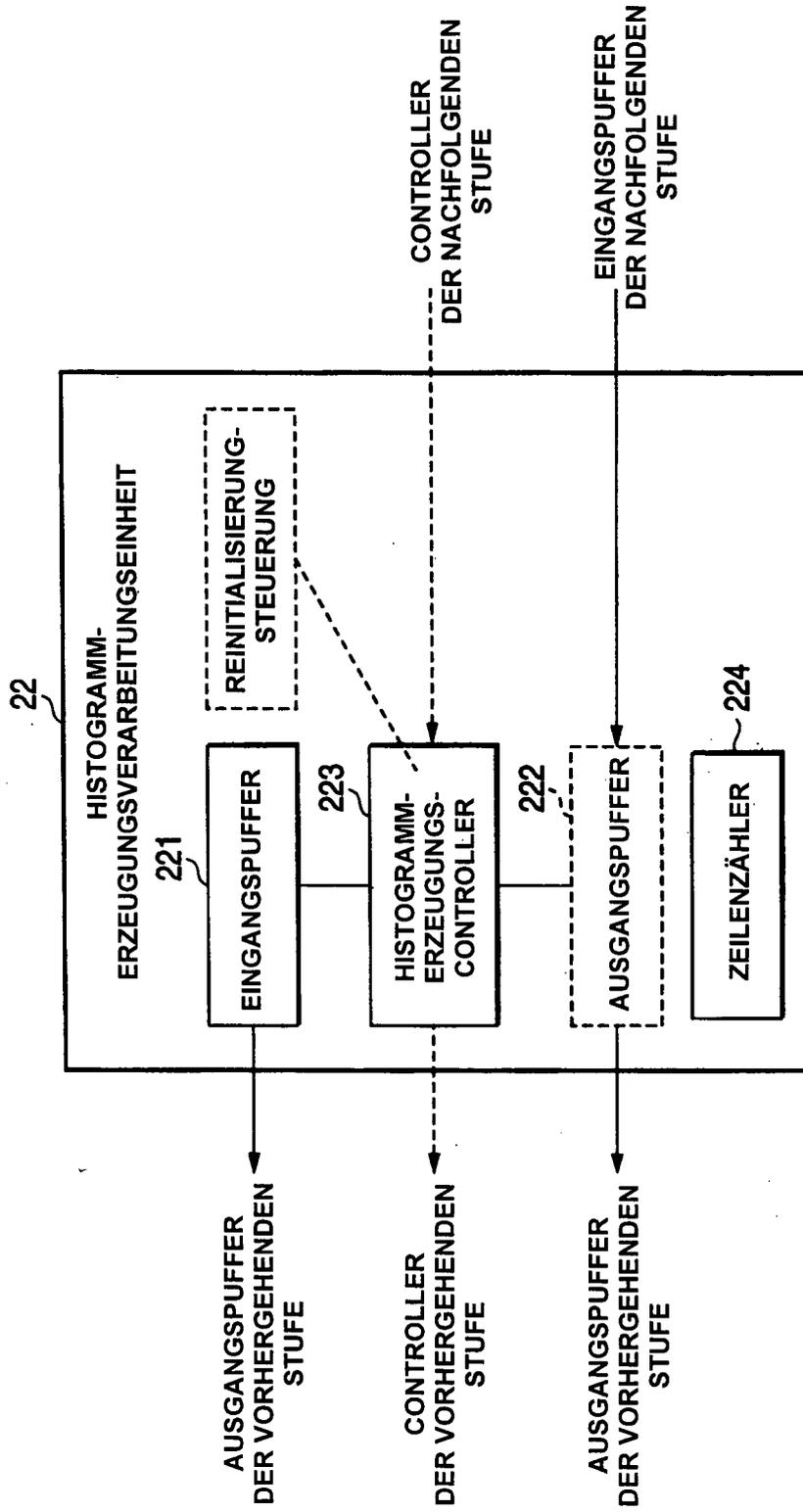


FIG. 5

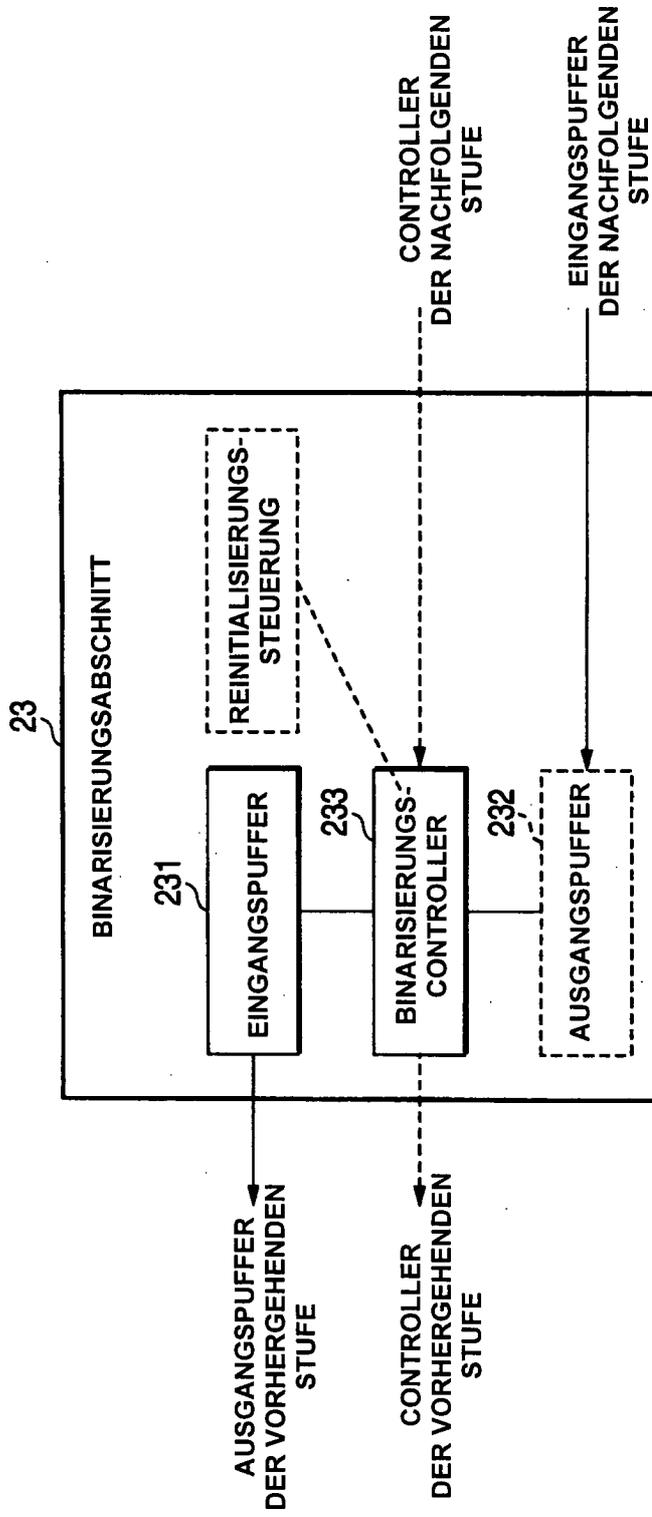


FIG. 6

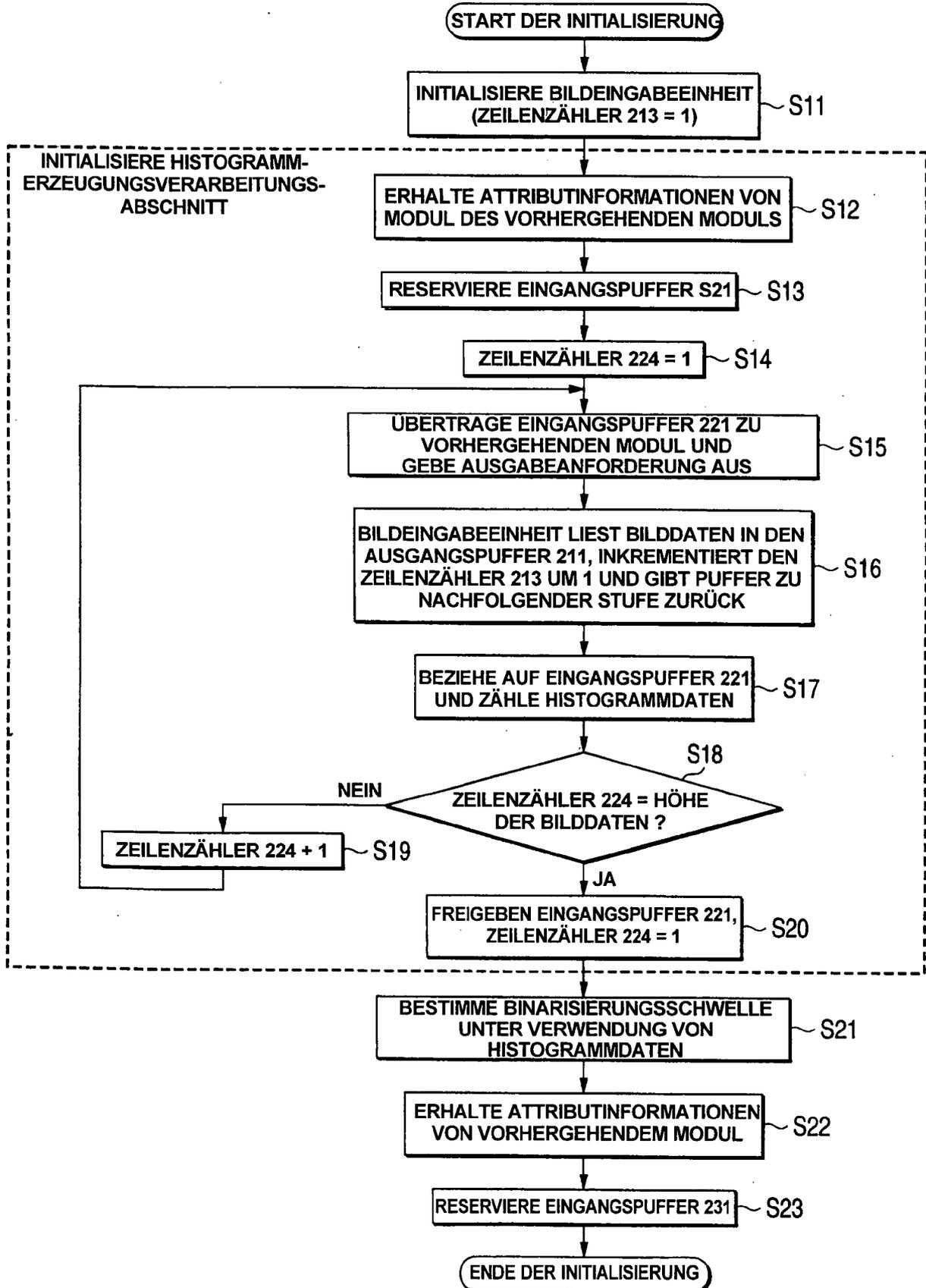


FIG. 7

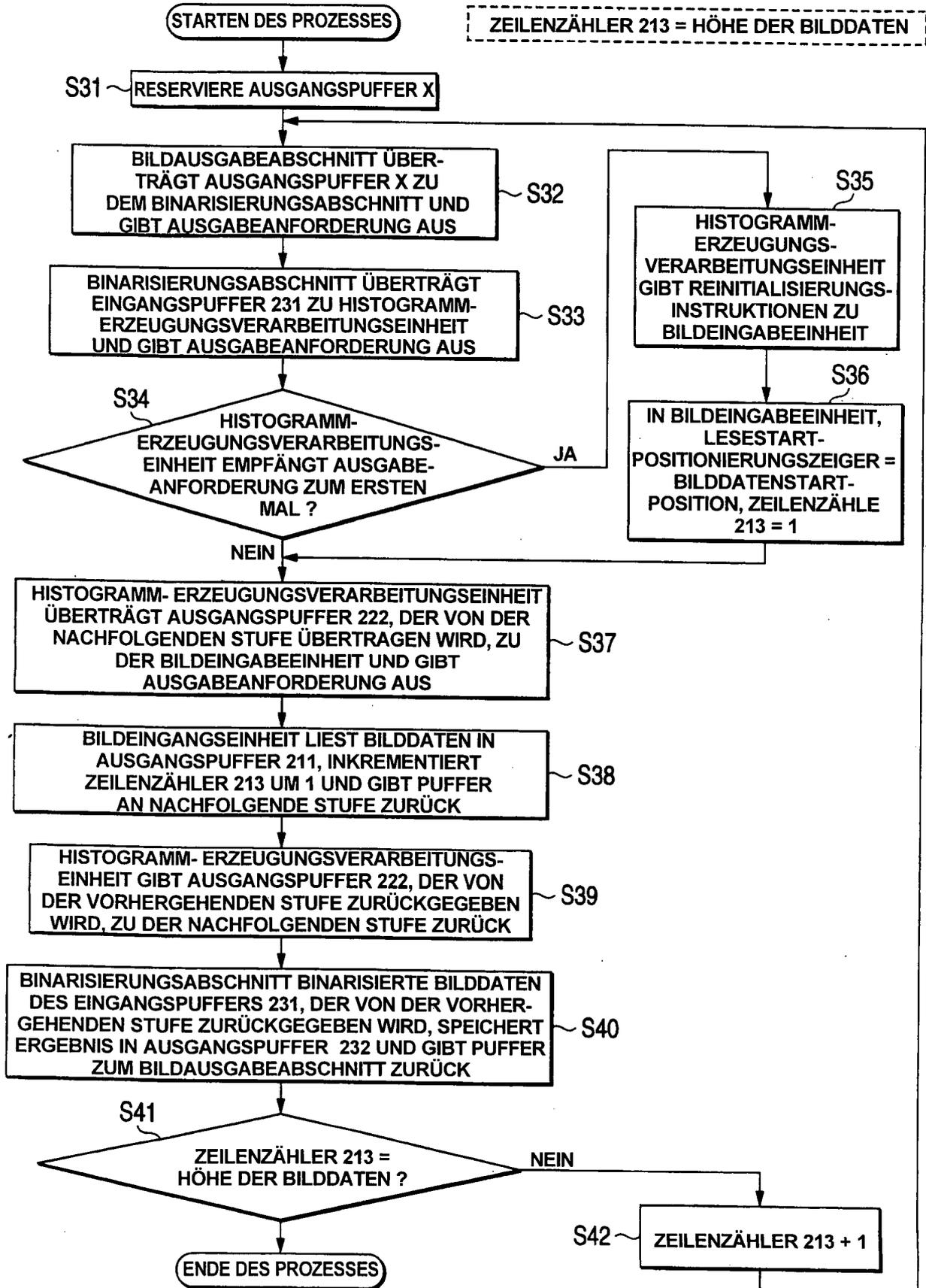


FIG. 8

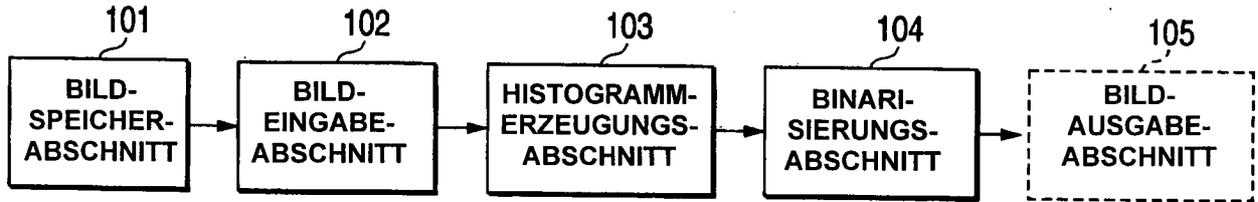


FIG. 9

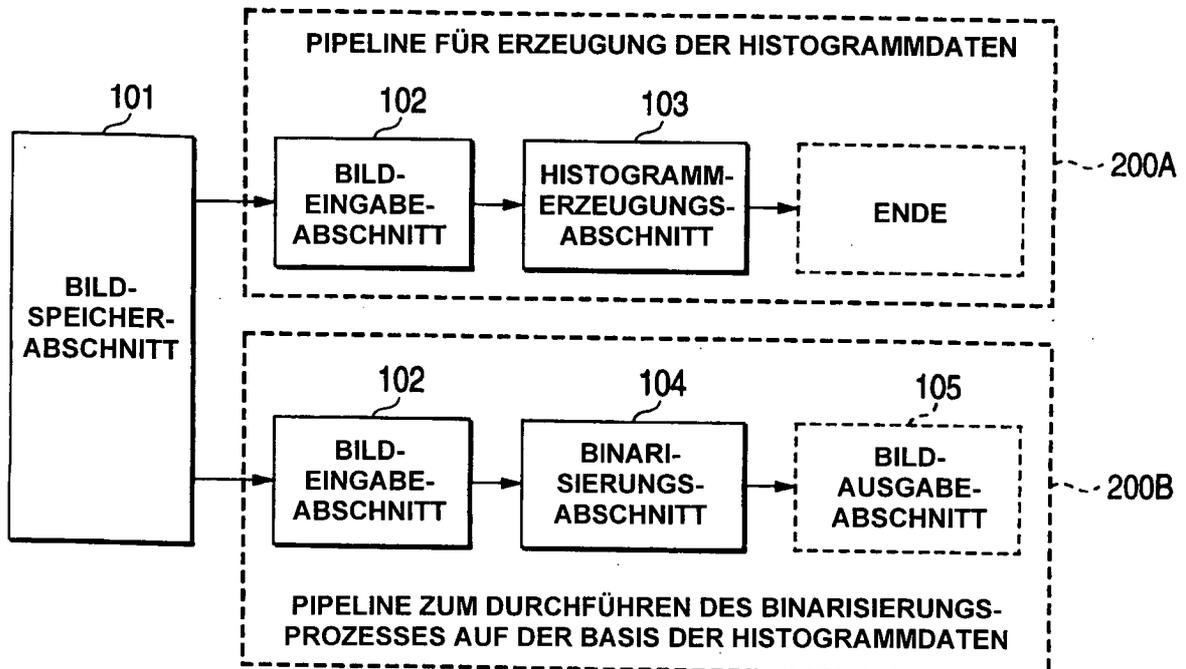


FIG. 10

