(19) **Bundesrepublik Deutschland**
Deutsches Patent- und Markenamt

(10) **DE 698 39 277 T2** 2009.04.30

(12) # Übersetzung der europäischen Patentschrift

(97) **EP 1 025 558 B1**
(21) Deutsches Aktenzeichen: **698 39 277.9**
(86) PCT-Aktenzeichen: **PCT/US98/01955**
(96) Europäisches Aktenzeichen: **98 903 891.4**
(87) PCT-Veröffentlichungs-Nr.: **WO 1998/036379**
(86) PCT-Anmeldetag: **28.01.1998**
(87) Veröffentlichungstag der PCT-Anmeldung: **20.08.1998**
(97) Erstveröffentlichung durch das EPA: **09.08.2000**
(97) Veröffentlichungstag der Patenterteilung beim EPA: **19.03.2008**
(47) Veröffentlichungstag im Patentblatt: **30.04.2009**

(51) Int Cl.[8]: ***G09G 5/02*** (2006.01)
G06T 15/50 (2006.01)
G06T 15/20 (2006.01)

(30) Unionspriorität:
794749 **03.02.1997** **US**

(84) Benannte Vertragsstaaten:
AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LI, LU, MC, NL, PT, SE

(73) Patentinhaber:
Micron Technology, Inc., Boise, Id., US

(72) Erfinder:
DONOVAN, Walter E., Milpitas, CA 95035, US

(74) Vertreter:
ZENZ Patent- und Rechtsanwälte, 45128 Essen

(54) Bezeichnung: **VERFAHREN UND ANORDNUNG ZUR AUSFÜHRUNG VON FARBSCHLÜSSELN, TRANSPARENZ UND NEBELFUNKTIONEN**

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

ERFINDUNGSGEBIET

[0001] Die Erfindung betrifft den Bereich der Computergrafik. Insbesondere betrifft die Erfindung ein Verfahren des Farbstanzens gefilterter Texturen mit und ohne Vernebeln, die auf einem Rastergrafikdisplay aufgezeichnet sind.

ALLGEMEINER STAND DER TECHNIK

[0002] Ein Computergrafikbild ist normalerweise aus einer Anzahl von Objekten aufgebaut. Objekte werden auf einem Hintergrund wiedergegeben. Anteile dieser Objekte können als transparent festgelegt werden, um ein Überlagern der Objekte auf eine Vielfalt später definierter Hintergrundbilder zu ermöglichen. Während der Wiedergabe kann das Objekt unter Verwendung von Zusammensetzungs-Techniken mit vorher erzeugten Objekten (z. B. zur Wiederverwendung eines komplexen Hintergrunds) kombiniert werden.

[0003] Das Zusammensetzen ist das Kombinieren von zwei Bildern, indem sie überlagert oder gemischt werden. In einem zusammengesetzten Bild wird der Wert eines jeden Pixels aus den Komponentenbildern berechnet. Ein Verfahren, das normalerweise verwendet wird, ist ein Überlagern, bei dem den Pixeln des Vordergrundbildes Transparenzwerte sowie beliebige andere Werte, die Pixel aufweisen können, wie z. B. Farbinformationen, gegeben werden müssen. Deshalb wird in dem zusammengesetzten Bild der Wert eines Pixels aus dem Hintergrundbild entnommen, außer wenn das Vordergrundbild an diesem Punkt einen nicht-transparenten Wert aufweist. Weist das Vordergrundbild an diesem Punkt einen nicht-transparenten Wert auf, dann wird der Wert aus dem Vordergrundbild entnommen. Es können auch mehrere Bilder gemischt werden, was zu Pixelwerte führt, die Linearkombinationen der Werte der mehreren Komponentenpixel sind.

[0004] Es treten jedoch Probleme auf, wenn ein Bild gezeichnet wird, das transparente Elemente enthält. Als ein Beispiel wird die Wiedergabe eines Bildes eines Baumes betrachtet. Obwohl nur die Wiedergabe des Baumes verlangt ist, wird das Baumbild normalerweise in einem rechteckigen Bild, nicht in einem baumförmigen Bild, gespeichert. Somit ist ein bestimmtes Verfahren erforderlich, die Teile des Bildes zu kennzeichnen, die nicht als ein Teil des Baumes anzusehen sind, und dann sollten diese Teile, die als die transparenten Teile zu bezeichnen sind, keinen Einfluss auf das aufgezeichnete Bild haben. Somit wird beim Aufzeichnen des Baumbildes die Anzeige-Hardware die nicht-transparenten Teile des Baums aufzeichnen und die transparenten Teile nicht aufzeichnen.

[0005] Zwei gebräuchliche Zusammensetztechniken, die nach dem Stand der Technik zur Wiedergabe eines Objekts auf einem transparenten Hintergrund verwendet werden, sind das Alpha-Kanal-Zusammensetzen und das Farbstanzen. Beim Alpha-Kanal-Zusammensetzen wird das Problem, ein transparentes Bild zu zeichnen, durch Speichern eines Alpha-Kanals gelöst, der einen Alpha-Wert verwendet, um die Transparenz eines jeden Pixels zu codieren. Dies erfordert jedoch das Bereitstellen von zusätzlichem Speicher über den für das Bild bereitgestellten hinaus. Das Farbstanzen erfordert jedoch keinen zusätzlichen Speicherplatz, da diese Technik die transparenten Bereiche des Bildes durch eine spezielle Farbe darstellt, die als die Chroma-Farbe bekannt ist. Beim Farbstanzen vergleicht die Hardware jedes Pixel mit der Chroma-Farbe, und wenn sie identisch sind, dann zeichnet sie das Pixel nicht auf dem Display auf. Diese Techniken wurden ausgebaut, um auf das Textur-Abbilden anwendbar zu sein. Anspruchsvolle Computergrafik-Wiedergaben enthalten normalerweise Muster oder Bilder, die auf eine Oberfläche abgebildet sind. Dieses Herangehen wird als Textur-Abbilden (texture mapping) bezeichnet; das Muster oder Bild wird als eine Textur-Abbildung bezeichnet, und seine einzelnen Elemente werden oft als Texel bezeichnet. Die Textur-Abbildung ist in ihrem eigenen (u, v)-Texturkoordinatenraum hinterlegt. Bei jedem wiedergegebenen Pixel werden ausgewählte Texel verwendet, um einen oder mehrere der Materialeigenschaften der Oberfläche, wie z. B. ihre diffusen Farbkomponenten, entweder zu ersetzen oder zu skalieren. Ein Pixel wird oft durch eine Anzahl von Texeln abgedeckt. Es sollte beachtet werden, dass das Farbstanzen, wie es oben beschrieben ist, nur funktioniert, wenn das Bild punktweise abgetastet ist, d. h., für jedes dargestellte Pixel wird ein Texel gelesen.

[0006] Die Wiedergabe nach dem Stand der Technik, soweit sie durch Farbstanzen erfolgt, wird normalerweise auf ungefilterte und nicht vernebelte Texturen angewendet. Die Textur-Abbildungen sind jedoch im Allgemeinen gefiltert, weil eine Verbesserung des Erscheinungsbildes von textur-abgebildeten Polygonen ein Kantenglätten des Textur-Abbildens erfordert. Es gibt viele gut bekannte Techniken für das Kantenglätten der Textur-Abbildung. Eine derartige Technik setzt das bilineare Filtern ein, das die Farben der vier nächsten Texel verwendet und die Farben zusammen wichtet, um eine gefilterte Darstellung dieser Textur zu erhalten. Für eine weitergehende Beschreibung des bilinearen Filterns siehe Wolberg, George, Digital Image Warping, IEEE Computer Society Press (1990), S. 57–61.

[0007] Es tritt jedoch ein Problem auf, wenn die Textur-Abbildung zu filtern ist. Statt ein einziges Texel zu einem Zeitpunkt zu lesen und dieses Texel zu verwenden, werden mehrere Texel gelesen und gefiltert,

und das gefilterte Ergebnis wird dann dargestellt. Das führt zu drei Szenarien: (a) alle gelesenen Texel sind die Chroma-Farbe; (b) keines der gelesenen Texel ist die Chroma-Farbe; oder (c) einige der gelesenen Texel sind die Chroma-Farbe. In dem Fall, in dem alle gelesenen Texel die Chroma-Farbe sind, wird das gefilterte Ergebnis generell die Chroma-Farbe sein, und beim Stand der Technik wird dieses Texel ordnungsgemäß nicht aufgezeichnet. In dem Fall, in dem keines der gelesenen Texel die Chroma-Farbe ist, sind keine transparenten Texel beteiligt, die Chroma-Farbe wird nicht ein wahrscheinliches Ergebnis des Filterns sein, und beim Stand der Technik wird dieses Texel ordnungsgemäß aufgezeichnet.

[0008] In dem Fall, in dem einige gelesene Texel die Chroma-Farbe sind, tritt das oben erwähnte Problem auf. In diesem Fall sollte das Texel als teilweise transparent angesehen werden (da es einige transparente Texel und einige opake Texel aufweist), aber beim Stand der Technik kommt es dazu, dass die Texel und die Chroma-Farbe zusammen gefiltert werden, und da es sehr unwahrscheinlich ist, dass die entstehende Farbe zur Chroma-Farbe passt, wird das gefilterte Ergebnis angezeigt. Somit verändert der Vorgang des Filterns die Farben am Rand der Textur-Abbildung, indem die Bildfarben mit den Farben der umliegenden Hintergrund-Texel vermischt werden, so dass die Hintergrund-Texel in der unmittelbaren Nachbarschaft des Bildes nicht länger zur Chroma-Farbe passen. Das führt dazu, dass um das wiedergegebene Bild herum eine Umrandung oder Kontur erscheint, so dass dann, wenn die Chroma-Farbe blau ist, wie es oft vorkommt, des entstehende Bild einen blauen Saum aufweisen wird. Folglich ist es erwünscht, ein Farbstanzen von gefilterten Texturen auszuführen.

[0009] Die Technik des "Vernebelns" oder des Hinzufügens eines Nebeleffekts zu einem dargestellten Bild ist in Videospielen beliebt, um Nebel- oder Rauch-Spezialeffekte zu erzeugen. Das Vernebeln fügt den Wiedergaben eine zusätzliche Stufe der Komplexität hinzu.

KURZDARSTELLUNG

[0010] Ein Verfahren, ein System und ein Schaltkreis zum Farbstanzen vernebelter und gefilterter Texturen werden in einem Computersystem bereitgestellt, wie in den angefügten Ansprüchen offengelegt ist. Gemäß einer Ausbildung der Erfindung wird ermittelt, ob ein Texel, das eine bestimmte Farbe aufweist, mit einer vorgegebenen Farbe übereinstimmt. Wurde ermittelt, dass die Texel-Farbe mit der vorgegebenen Farbe übereinstimmt, dann wird das Texel durch ein Texel einer anderen Farbe ersetzt. Die Texel werden dann gefiltert. Die farbgestanzten und gefilterten Texel werden dann vernebelt. Die farbgestanzten, vernebelten und gefilterten Texel werden mit einer Hintergrundfarbe gemischt.

[0011] In einer Ausführungsform enthält die Vorrichtung im Allgemeinen einen Prozessor, einen an den Prozessor gekoppelten Speicher und einen Co-Prozessor, der an Prozessor und den Speicher gekoppelt ist. Der Co-Prozessor ermittelt, ob ein Texel, das eine bestimmte Farbe aufweist, mit einer vorgegebenen Farbe übereinstimmt. Wenn ermittelt wurde, dass die Texel-Farbe mit der vorgegebenen Farbe übereinstimmt, dann ersetzt der Co-Prozessor das Texel durch ein Texel einer anderen Farbe. Der Co-Prozessor filtert dann die Texel. Der Co-Prozessor enthält die Schaltung für das Vernebeln der farbgestanzten und gefilterten Texel. Die farbgestanzten, vernebelten und gefilterten Texel werden mit einer Hintergrundfarbe gemischt.

[0012] Diese und andere Merkmale, Gesichtspunkte und Vorteile der vorliegenden Erfindung werden aus den beigefügten Zeichnungen und aus der ausführlichen Beschreibung und den angefügten Ansprüchen, welche nachfolgen, ersichtlich.

KURZBESCHREIBUNG DER ZEICHNUNGEN

[0013] Die vorliegende Erfindung wird als Beispiel und nicht als Einschränkung in den Figuren der beigefügten Zeichnungen erläutert, in denen gleiche Bezugsziffern ähnliche Elemente bezeichnen und in denen:

[0014] Fig. 1 eine Ausführungsform eines Computersystems darstellt, in dem die vorliegende Erfindung implementiert ist.

[0015] Fig. 2 eine Ausführungsform einer Grafik/Video-Verarbeitungsschaltung darstellt, in der die vorliegende Erfindung implementiert ist.

[0016] Fig. 3 eine Ausführungsform einer Pixel-Engine darstellt, in der die vorliegende Erfindung implementiert ist.

[0017] Fig. 4 eine Ausführungsform einer Pixel-Pipeline darstellt, in der die vorliegende Erfindung implementiert ist.

[0018] Fig. 5 ein Flussdiagramm zum Realisieren einer Ausführungsform eines Prozesses für das Farbstanzen von vernebelten, gefilterten Texturen darstellt.

AUSFÜHRLICHE BESCHREIBUNG

[0019] Es wird ein Verfahren und eine Vorrichtung zum Farbstanzen gefilterter Texturen in einem Computersystem bereitgestellt. In der nachfolgenden Beschreibung werden zu Erläuterungszwecken viele spezifische Details dargestellt, um für ein gründliches

Verständnis der vorliegenden Erfindung zu sorgen. Für eine Fachperson wird es jedoch offensichtlich sein, dass die vorliegende Erfindung ohne diese spezifischen Details realisiert werden kann. In anderen Fällen werden gut bekannte Strukturen und Geräte in einer Blockdiagrammform gezeigt, um ein unnötiges Verschleiern der vorliegenden Erfindung zu vermeiden.

[0020] Fig. 1 stellt eine Ausführungsform eines Computersystems **1** dar, in dem die vorliegende Erfindung implementiert ist. Das Computersystem **1** enthält eine Zentraleinheit (CPU) **10**, die über einen Systembus **30** an den Systemspeicher **20** angeschlossen ist. Die CPU **10** und der Speicher **20** sind über den Systembus **30** durch eine Bus-Schnittstelle **50** an einen PCI(peripheral component interconnect)-Bus **40** angeschlossen. An den PCI-Bus **40** sind eine Grafik/Video-Beschleunigerkarte **60** sowie verschiedene periphere Geräte **80** und **90** angeschlossen. Die Grafik/Video-Beschleunigerkarte **60** ist an ein Bildschirmanzeigegerät **70** angeschlossen.

[0021] Fig. 2 stellt ein Beispiel für eine Schaltung dar, die in der Grafik/Video-Beschleunigerkarte **60** enthalten ist, die eine Schaltung zum Ausführen verschiedener dreidimensionaler (3D) Grafikfunktionen umfasst. In Fig. 2 verbindet eine PCI-Schnittstelle **100** die Grafik/Video-Beschleunigerkarte **60** mit dem PCI-Bus **40**. An die PCI-Schnittstelle **100** ist ein Grafikprozessor **102** angeschlossen, und er ist für die Ausführung verschiedener Grafik- und Videoverarbeitungsfunktionen ausgelegt. Der Grafikprozessor **102** ist normalerweise ein RISC(reduced instruction set computing)-Processor.

[0022] An den Grafikprozessor **102** ist eine Pixel-Engine **120** angeschlossen, und sie enthält Schaltungen zum Ausführen verschiedener Grafikfunktionen, wie z. B. das bilineare Filtern, Vernebeln und Mischen, wie nachfolgend beschrieben wird. Ein lokaler Speicher mit wahlfreiem Zugriff (RAM) **110** speichert sowohl die Quellenpixel-Farbwerte als auch die Zielpixel-Farbwerte. Die Zielfarbwerte werden in einem Rahmenspeicher **112** innerhalb des Speichers **110** gespeichert. In der bevorzugten Ausführungsform ist der Speicher **110** unter Verwendung eines dynamischen RAM (DRAM) implementiert. An den RAM **110** und an einen First-In-First-Out(FIFO)-Pufferspreicher 116 ist eine Anzeigesteuereinheit **114** angeschlossen. Gesteuert durch die Anzeigesteuereinheit **114** werden die im Rahmenspeicher **112** gespeicherten Zielfarbwerte dem FIFO **116** bereitgestellt. Die im FIFO **116** gespeicherten Zielwerte werden einem Satz von Digital-Analog-Wandlern (DAW) **118** zur Verfügung gestellt, die zum Bildschirm **70** rote, grüne und blaue Analog-Farbsignale ausgeben.

[0023] An den RAM **110** ist auch eine Speichersteuereinheit **108** angeschlossen. Die Speichersteuereinheit **108** steuert die Übertragung von Daten zwischen dem RAM **110** und der Pixel-Engine **120** als auch dem Grafikprozessor **102**. Ein Anweisungs-Cachespeicher **104** und ein Daten-Cachespeicher **106** sind beide an den Grafikprozessor **102** und die Speichersteuereinheit **108** angeschlossen und werden verwendet, um häufig benutzte Anweisungen bzw. Daten zu speichern. Der Daten-Cachespeicher **106** ist auch an die PCI-Schnittstelle **100** und die Pixel-Engine **120** angeschlossen.

[0024] Fig. 3 stellt eine Ausführungsform der Pixel-Engine **120** detaillierter dar. Die Pixel-Engine **120** enthält eine Befehls-Warteschlange **130**, eine Leseanforderungs-Warteschlange **134**, ein Attribut-Warteschlange **136**, eine Lesedaten-Warteschlange **138**, eine Schreibdaten-Warteschlange **142**, eine Stufe-1-Verarbeitungsschaltung **132** und eine Stufe-2-Verarbeitungsschaltung **140**. Die vom Grafikprozessor **102** ausgegebenen Befehle werden von der Pixel-Engine **120** empfangen und in einer Befehls-Warteschlange **130** gespeichert. Die Befehle, die in der Befehls-Warteschlange **130** gespeichert sind, werden dann der Stufe-1-Schaltung **132** zur Verfügung gestellt. Die Stufe-1-Schaltung **132** sammelt die Pixelattribute und bestimmt die Anzahl der Lese- und Schreibzyklen, die für jeden Befehl auszuführen sind. Die Pixelattribute können zum Beispiel x-, y- und z-Komponenten, R-, G- und B-Komponenten, Alpha(Transparenz)-, u- und v-Komponenten und den Nebel enthalten. Außerdem erzeugt die Stufe-1-Schaltung **132** Speicherlese- und -schreibanforderungen und Adressen, die mit jedem Lese- und Schreibvorgang verbunden sind. Die durch die Stufe-1-Schaltung **132** erzeugten Speicherleseanforderungen werden in der Leseanforderungs-Warteschlange **134** gespeichert. Die Leseanforderungs-Warteschlange **134** gibt dann jede Leseanforderung an den Daten-Cachespeicher **106** aus. Außerdem gibt die Stufe-1-Schaltung **132** Pixelattribute an die Attribut-Warteschlange **136** aus, welche die Attribute anschließend der Stufe-2-Schaltung **140** zur Verfügung stellt.

[0025] Während der Lesevorgänge werden die Pixel-Farbwerte aus dem Daten-Cachespeicher **106** (die Farbwerte können aus dem RAM **110** stammen) in die Lesedaten-Warteschlange **138** gelesen. Die Pixel-Farbwerte werden dann aus der Lesedaten-Warteschlange **138** in die Stufe-2-Schaltung **140** ausgegeben, welche Funktionen ausführt, die bilineares Filtern, Texturanwendung, Nebeleffekt, Farbraumumwandlung, Mischen und Dithern einschließen. Die verarbeiteten Farbwerte werden dann durch die Stufe-2-Schaltung **140** einer Schreibdaten-Warteschlange **142** zur Verfügung gestellt, welche dann die verarbeiteten Farbwerte dem Daten-Cachespeicher **106** ausgibt.

[0026] Die Attribut-Warteschlange **136** speichert Pixelattribute, die nicht verwendet werden, um Daten im Daten-Cachespeicher **106** oder RAM **110** aufzusuchen. Wenn Daten aus dem Daten-Cachespeicher **106** in die Stufe-2-Schaltung **140** gelesen werden, dann werden die entsprechenden Attributwerte, die in der Attribut-Warteschlange **136** gespeichert sind, parallel aus der Attribut-Warteschlange **136** zur Stufe-2-Schaltung **140** ausgelesen.

[0027] In einer Ausführungsform ist die vorliegende Erfindung in der Schaltung innerhalb der Stufe-2-Schaltung **140** implementiert. Wie für eine Fachperson leicht ersichtlich ist, könnte der Prozess in anderen Teilen des Schaltkreises implementiert sein. Fig. 4 stellt eine Ausführungsform der Stufe-2-Schaltung **140** ausführlicher dar. Die Stufe-2-Schaltung **140** implementiert eine Pixel-Pipeline, in der die folgenden Funktionen ausgeführt werden: Pixel-Entpacken, Farbstanzen, bilineares Filtern und Texturanwendung, Vernebeln und Mischen. Diese Funktionen werden durch eine Zustandsmaschine **146** koordiniert und terminiert. Es wird in Betracht gezogen, dass die Stufe-2-Schaltung weniger als all die oben bezeichneten Funktionen aufweisen kann.

[0028] In der Stufe-2-Schaltung **140** empfängt ein Entpack-Schaltkreis **148** über das Signal READ DATA die Pixel-Farbwerte aus dem RAM **110** oder dem Daten-Cachespeicher **106**, und er empfängt über das Signal ATT DATA die Attributwerte. Der Entpack-Schaltkreis **148** wandelt die Farbdaten in Segmente von 8 Bits um. Zum Beispiel könnten die Farbdaten dem Entpack-Schaltkreis **148** in 16-Bit-Wörtern bereitgestellt werden, wobei jedes Wort zum Beispiel aus drei Fünf-Bit-Farbsegmenten und 1 ungenutzten Bit besteht. Dementsprechend würde der Entpack-Schaltkreis **148** jedes Fünf-Bit-Farbsegment in einem jeden 16-Bit-Wort in einen getrennten 8-Bit-Farbwert umwandeln ("entpacken"). Die "entpackten" Farbwerte werden dann durch den Entpack-Schaltkreis **148** der Farbstanz-Schaltung **149** zugeführt.

[0029] In einer Ausführungsform der vorliegenden Erfindung funktioniert die Farbstanz-Schaltung **149** durch Vergleich der Farbe eines jeden eingegebenen Texel-Signals mit einer Chroma-Farbe, die in der Farbstanz-Schaltung **149** gespeichert ist. Wenn die Farbstanz-Schaltung **149** im RGB-Farbraum feststellt, dass die Farbe des eingegebenen Texels die gleiche ist wie die gespeicherte Chroma-Farbe, dann weist das Texel, das durch die Farbstanz-Schaltung **149** ausgegeben wird, die Signalkomponentenwerte (A, R, G, B) von jeweils 0, 0, 0 und 0 auf, was dem Schwarz entspricht. Im RGB-Farbraum stellen die Komponenten A, R, G und B jeweils die Alpha-, Rot-, Grün- und Blau-Komponenten des Farbsignals dar. Deshalb wird dann, wenn die Farbe des eingegebenen Texels die gleiche ist wie die Chroma-Farbe ist, wie durch den Chroma-Farbe-Schaltkreis **149** festgestellt wurde, ein transparentes Texel ausgegeben. Dieser Vorgang ist nicht auf den RGB-Farbraum beschränkt, und die Farbstanz-Schaltung **149** kann bei einem Nachweis, dass die Farbe des eingegebenen Texels die gleiche ist wie eine gespeicherte Chroma-Farbe, zur Ausgabe von Signalkomponentenwerten veranlasst werden, die Schwarz kennzeichnen, was in einem speziellen Farbraum einem transparenten Texel entspricht.

[0030] Wenn die Farbstanz-Schaltung **149** feststellt, dass die Farbe des eingegebenen Texels nicht die gleiche ist wie die gespeicherte Chroma-Farbe, dann weist das Texel, das durch die Farbstanz-Schaltung **149** ausgegeben wird, die Signalkomponentenwerte (A, R, G, B) von jeweils 1, R, G und B auf. Somit wird dann, wenn sich die Farbe des eingegebenen Texels von der Chroma-Farbe unterscheidet, wie durch den Chroma-Farbe-Schaltkreis **149** festgestellt wurde, das ausgegebene Texel die gleichen Rot-, Grün- und Blau-Signalkomponenten wie das eingegebene Texel aufweisen, wobei die Alpha-Signalkomponente gleich 1 gesetzt wird. Das ergibt ein opakes Ausgabe-Texel, das die gleiche Farbe aufweist wie das eingegebene Texel. Die farbgestanzten Texel werden dann durch die Farbstanz-Schaltung **149** der Schaltung für die bilineare Filterung und Texturanwendung **150** zur Verfügung gestellt.

[0031] Das Verbessern des Aussehens texturierter Oberflächen erfordert ein kantenglättendes Textur-Abbilden. Die Bilinearfilterungs-Schaltung **150** führt Funktionen aus, um die Anwendung texturierter Oberflächen auf ein Bild zu verbessern. In einer Ausführungsform der vorliegenden Erfindung wird das bilineare Filtern verwendet. Die Bilinearfilterungs-Schaltung **150** nimmt die Farbe der vier nächsten Texel auf und wichtet die Farben zusammen, um eine gefilterte Darstellung dieser Textur zu erhalten. Wenn das bilineare Filtern angewendet wird, dann ist das ausgegebene Texel eine Funktion von vier angrenzenden eingegebenen Texeln und kann als eine 2 × 2-Faltung mit Wichtungen angesehen werden, wie sie nachfolgend dargestellt sind:

$f_1 f_2$ $f_1(1 - f_2)$

$(1 - f_1)f_2$ $(1 - f_1)(1 - f_2)$

wobei f_l und f_2 im Bereich von 0 bis 1 einschließlich liegen.

[0032] Im RGB-Farbraum gibt die Bilinearfilterungs-Schaltung **150** ein gefiltertes Farbsignal (A, R, G, B) aus, das die gefilterten Alpha-, Rot-, Grün-, und Blau-Komponenten beinhaltet. Die Bilinearfilterungs-Schaltung **150** gibt die Pixel-Farbwerte A, R, G, B an die Nebel-Schaltung **152** aus.

[0033] Die Nebel-Schaltung **152** wendet den Nebel auf ausgewählte gefilterte Texel an, bei denen der Nebel auf einem angezeigten Bild zu erscheinen hat. Der Nebel wird durch ein Farbsignal dargestellt, welches die Komponenten Rfog, Gfog und Bfog aufweist, die jeweils die rote, grüne und blaue Komponente des Nebelsignals darstellen. In einer Ausführungsform der vorliegenden Erfindung wird ein vernebeltes und gefiltertes Signal (A1, R1, G1, B1) für jede Komponente des Farbsignals (A, R, G, B) gemäß den Gleichungen (1)–(4) erzeugt.

$$A1 = A \quad (1)$$

$$R1 = [R\cdot(1 - f)] + (Rfog\cdot f\cdot A) \quad (2)$$

$$G1 = [G\cdot(1 - f)] + (Gfog\cdot f\cdot A) \quad (3)$$

$$B1 = [B\cdot(1 - f)] + (Bfog\cdot f\cdot A) \quad (4)$$

[0034] In den Gleichungen (1)–(4) stellt A die Alpha-Komponente des gefilterten Signals aus der Bilinearfilterungs-Schaltung **150**, R die Rot-Komponente des gefilterten Farbsignals aus der Bilinearfilterungs-Schaltung **150**, Rfog die Rot-Komponente der Nebelfarbe, f den Nebelfaktor, G die Grün-Komponente des gefilterten Farbsignals aus der Bilinearfilterungs-Schaltung **150**, Gfog die Grün-Komponente der Nebelfarbe, B die Blau-Komponente des gefilterten Farbsignals aus der Bilinearfilterungs-Schaltung **150** und Bfog die Blau-Komponente der Nebelfarbe dar. Der Nebelfaktor (f), der die Stärke oder den Prozentsatz der Nebelverdeckung darstellt, kann einen Wert aufweisen, der im Bereich von 0 bis 1 einschließlich liegt. Wenn f gleich 0 ist, dann wird kein Nebel dargestellt; wenn f gleich 1 ist, dann wird nur Nebel dargestellt. Die Nebel-Schaltung **152** gibt der Misch-Schaltung **154** die Farbwerte (A1, R1, G1, B1) aus.

[0035] Die Gleichungen (1)–(4) sind für die Verwendung im RGB-Farbraum vorgesehen, die Gleichungen (2)–(4) werden jedoch in der Gleichung (5) für die Anwendung in anderen Farbräumen verallgemeinert.

$$Xn1 = [Xn\cdot(1 - f)] + (Xnfog\cdot f\cdot A) \quad (5)$$

[0036] In der Gleichung (5) stellt A die Alpha-Komponente des gefilterten Signals dar, und f stellt wie in den Gleichungen (1)–(4) den Nebelfaktor dar. Die Variable Xn stellt eine spezielle Komponente eines gefilterten Farbsignals dar, das n Farbkomponenten ausweist, und Xnfog stellt die entsprechende Farbkomponente der Nebelfarbe dar. Bei einer Anwendung auf andere Farbräume gibt die Nebel-Schaltung **152** einen Wert für die Alpha-Komponente des Eingabesignals gemäß Gleichung (1) aus, und sie gibt Farbwerte für jede der n Farbkomponenten des Eingabesignal-Farbraumes gemäß Gleichung (5) aus.

[0037] Die Misch-Schaltung **154** mischt die gefilterte Texel-Farbe mit einer Hintergrundfarbe, um die Farbe des Ausgabe-Texels zu erzeugen. Die Hintergrundfarbe wird durch ein Farbsignal mit den Komponenten Ab, Rb, Gb und Bb dargestellt, die jeweils die Alpha-, Rot-, Grün und Blau-Komponente des Hintergrundsignals darstellen. Wurde ein gefiltertes Texel nicht vernebelt (f = 0), dann erzeugt die Misch-Schaltung **154** ein Ausgabe-Farbsignal (Ao, Ro, Go, Bo) gemäß den Gleichungen (6) bis (9).

$$Ao = A + [(1 - A)\cdot Ab] \quad (6)$$

$$Ro = R + [(1 - A)\cdot Rb] \quad (7)$$

$$Go = G + [(1 - A)\cdot Gb] \quad (8)$$

$$Bo = B + [(1 - A)\cdot Bb] \quad (9)$$

[0038] In den Gleichungen (6)–(9) stellt A die Alpha-Komponente des gefilterten Signals aus der Bilinearfilterungs-Schaltung **150**, Ab die Alpha-Komponente der Hintergrundfarbe, R die Rot-Komponente des gefilterten Farbsignals aus der Bilinearfilterungs-Schaltung **150**, Rb die Rot-Komponente der Hintergrundfarbe, G die Grün-Komponente des gefilterten Farbsignals aus der Bilinearfilterungs-Schaltung **150**, Gb die Grün-Komponente der Hintergrundfarbe, B die Blau-Komponente des gefilterten Farbsignals aus der Bilinearfilterungs-Schaltung **150** und Bb die Blau-Komponente der Hintergrundfarbe dar.

[0039] Die Gleichungen (6)–(9) sind für die Verwendung im RGB-Farbraum vorgesehen, die Gleichungen (7)–(9) werden jedoch in der Gleichung (10) für die Anwendung in anderen Farbräumen verallgemeinert.

$$Xno = Xn + [(1 - A)\cdot Xnb] \quad (10)$$

[0040] In der Gleichung (10) stellt A die Alpha-Komponente des gefilterten Signals wie in den Gleichungen (6)–(9) dar. Die Variable Xn stellt eine spezielle Komponente eines gefilterten Farbsignals dar, das n Farbkomponenten ausweist, und Xnb stellt die entsprechende Farbkomponente der Hintergrundfarbe dar. Bei einer Anwendung auf andere Farbräume gibt die Misch-Schaltung **154** einen Wert für die Alpha-Komponente des Eingabesignals gemäß Gleichung (6) aus, und sie gibt Farbwerte für jede der n Farbkomponenten des Eingabesignal-Farbraumes gemäß Gleichung (10) aus.

[0041] Die Misch-Schaltung **154** mischt eine vernebelte und gefilterte Texel-Farbe mit einer Hintergrundfarbe, um die Farbe des Ausgabe-Texels zu erzeugen. Für den Fall eines vernebelten und gefilterten Texels wird das Ausgabe-Farbsignal (Ao, Ro, Go, Bo) gemäß den Gleichungen (11) bis (14) durch die Misch-Schaltung **154** erzeugt.

$$Ao = A1 + [(1 - A1)\cdot Ab] \quad (11)$$

$$Ro = R1 + [(1 - A1)\cdot Rb] \quad (12)$$

$$Go = G1 + [(1 - A1)\cdot Gb] \quad (13)$$

$$Bo = B1 + [(1 - A1)\cdot Bb] \quad (14)$$

[0042] In den Gleichungen (11–(14) stellt A1 die Alpha-Komponente des vernebelten und gefilterten Signals aus der Nebel-Schaltung **152** (Gleichung 1), Ab die Alpha-Komponente der Hintergrundfarbe, R1 die Rot-Komponente des vernebelten und gefilterten Farbsignals aus der Nebel-Schaltung **152** (Gleichung 2), Rb die Rot-Komponente der Hintergrundfarbe, G1 die Grün-Komponente des vernebelten und gefilterten Farbsignals aus der Nebel-Schaltung **152** (Gleichung 3), Gb die Grün-Komponente der Hintergrundfarbe, B1 die Blau-Komponente des vernebelten und gefilterten Farbsignals aus der Nebel-Schaltung **152** (Gleichung 4) und Bb die Blau-Komponente der Hintergrundfarbe dar.

[0043] Die Gleichungen (11)–(14) sind für die Verwendung im RGB-Farbraum vorgesehen, die Gleichungen (12)–(14) werden jedoch in der Gleichung (15) für die Anwendung in anderen Farbräumen verallgemeinert.

$$Xno = Xn1 + [(1 - A1)\cdot Xnb] \quad (15)$$

[0044] In der Gleichung (15) stellt A1 die Alpha-Komponente des vernebelten und gefilterten Signals wie in den Gleichungen (11)–(14) dar. Die Variable Xn1 stellt eine spezielle Komponente des vernebelten und gefilterten Farbsignals dar, das n Farbkomponenten aufweist, und Xnb stellt die entsprechende Farbkomponente der Hintergrundfarbe dar. Bei einer Anwendung auf andere Farbräume gibt die Misch-Schaltung **154** einen Wert für die Alpha-Komponente des Eingabesignals gemäß Gleichung (11) aus, und sie gibt Farbwerte für jede der n Farbkomponenten des Eingabesignal-Farbraumes gemäß Gleichung (15) aus.

[0045] Fig. 5 stellt ein Flussdiagramm einer Ausführungsform zum Vollziehen des Farbstanzens von vernebelten, gefilterten Texturen im RGB-Raum dar. Das Flussdiagramm beginnt beim Block **300**. Der Ablauf geht weiter beim Block **302**, an dem die Farbe eines jeden eingegebenen Texels mit einer gespeicherten Chroma-Farbe verglichen wird. Wenn festgestellt wird, dass die Farbe des eingegebenen Texels die gleiche ist wie die gespeicherte Chroma-Farbe, dann geht der Ablauf beim Block **304** weiter, wo das eingegebene Texel durch ein Texel ersetzt wird, das eine andere Farbe aufweist als das eingegebene Texel. In einer Ausführungsform der vorliegenden Erfindung weist das ausgegebene Texel die Signalkomponentenwerte (A, R, G, B) von jeweils 0, 0, 0 und 0 auf. Deshalb wird dann, wenn die Farbe des eingegebenen Texels die gleiche ist wie die Chroma-Farbe, ein transparentes Texel ausgegeben.

[0046] Ein Farbmodell ist eine Spezifikation eines dreidimensionalen Farb-Koordinatensystems. Die hier beschriebene Ausführungsform der vorliegenden Erfindung arbeitet mit dem Farbmodell Rot, Grün und Blau (RGB). Eine alternative Ausführungsform der vorliegenden Erfindung arbeitet mit Signalen im YCBCR- oder YUV-Farbmodell. In einer 8-Bit-Version dieser alternativen Ausführungsform arbeitet die Farbstanz-Schaltung **149** durch Vergleich der Farbe eines jeden eingegebenen Texelsignals mit einer Chroma-Farbe, die in der Farbstanz-Schaltung **149** gespeichert ist. Stellt die Farbstanz-Schaltung **149** fest, dass die Farbe des eingegebenen Texels die gleiche ist wie die gespeicherte Chroma-Farbe, dann weist das Texel, das durch die Farbstanz-Schaltung **149** ausgegeben wird, die Signalkomponentenwerte (A, CR, Y, CB) von jeweils 0, 128, 16 und 128 auf, um im YCBCR-Farbraum dem Schwarz zu entsprechen. Deshalb wird dann, wenn die Farbe des eingegebenen Texels die gleiche ist wie die Chroma-Farbe, wie durch die Farbstanz-Schaltung **149** festgestellt wurde, ein transparentes Texel ausgegeben. Außerdem können das Verfahren und die Vorrichtung, die hier beschrieben sind, auf einen beliebigen Farbraum angewendet werden, für den eine lineare Beziehung zum RGB- oder YCBCR-Farbraum besteht oder der unter Verwendung einer linearen Transformation des RGB- oder YCBCR-Farbraums erhalten werden kann.

[0047] Mit dem Beispiel des RGB-Raumes fortfahrend wird dann, wenn festgestellt wird, dass die Farbe des eingegebenen Texels nicht die gleiche ist wie die gespeicherte Chroma-Farbe, der Ablauf am Block **306** fortgesetzt, wobei das ausgegebene Texel die gleiche Farbe aufweist wie das eingegebene Texel. In einer Ausführungsform der vorliegenden Erfindung weist das ausgegebene Texel die Signalkomponentenwerte (A, R, G, B) von jeweils 1, R, G und B auf. Wenn sich die Farbe des eingegebenen Texels von der Chroma-Farbe unterscheidet, dann wird somit das ausgegebene Texel die gleichen Rot-, Grün- und Blau-Signalkomponenten wie das eingegebene Texel aufweisen, wobei die Alpha-Signalkomponente gleich 1 gesetzt wird. Das ergibt ein opakes Texel, das die gleiche Farbe wie das eingegebene Texel aufweist. In der alternativen Ausführungsform der vorliegenden Erfindung, in der das YCBCR-Farbmodell verwendet wird, weist das ausgegebene Texel die Signalkomponentenwerte (A, CR, Y, CB) von jeweils 1, CR, Y und CB auf.

[0048] Die Verbesserung des Erscheinungsbildes von textur-abgebildeten Polygonen erfordert ein kantenglättendes Textur-Abbilden. Deshalb wird anschließend an die Farbstanz-Vorgänge entweder

vom Block **304** oder vom Block **306** der Ablauf am Block **308** fortgesetzt, an dem alle Texel, die vom Farbstanzen ausgegeben wurden, unter Verwendung eines bilinearen Filterns gefiltert werden. Es gibt viele gut bekannte Verfahren für das Kantenglätten der Textur-Abbildung. Die bevorzugte Ausführungsform der vorliegenden Erfindung verwendet das bilineare Filtern. Das bilineare Filtern nimmt die Farbe der vier nächsten Texel auf und wichtet die Farben zusammen, um eine gefilterte Darstellung dieser Textur zu ermitteln. Wenn ein bilineares Filtern angewendet wird, dann ist das ausgegebene Pixel eine Funktion von vier benachbarten Eingabe-Pixeln. Das Filtern der Texel ergibt ein gefiltertes Farbsignal, das sich aus gefilterten Alpha-, Rot-, Grün- und Blau-Komponenten zusammensetzt.

[0049] Anschließend an das Filtern geht der Ablauf am Block **310** weiter, an dem die Rot-, Grün- und Blau-Komponenten des gefilterten Signals vernebelt werden. Der Nebel wird gemäß den Gleichungen (1)–(4) erzeugt. in den Gleichungen (1)–(4) stellen R, G, B und A jeweils die Rot-, Grün-, Blau- und Alpha-Komponente des gefilterten Farbsignals aus dem Block **308** dar. Rfog, Gfog und Bfog stellen jeweils die Rot-, Grün- und Blau-Komponenten der Nebelfarbe dar, und f stellt den Nebelfaktor dar. Der Nebelfaktor (f), der die Stärke oder den Prozentsatz der Nebelverdeckung darstellt, kann einen Wert aufweisen, der im Bereich von 0 bis 1 einschließlich liegt.

[0050] Wenn f gleich 0 ist, dann wird kein Nebel dargestellt, wenn f gleich 1 ist, dann wird nur Nebel dargestellt.

[0051] Anschließend an das Vernebeln geht der Ablauf am Block **312** weiter, an dem die vernebelten, gefilterten Texel mit einer Hintergrundfarbe gemischt werden, um ein Ausgabe-Farbsignal zu erzeugen. Das aus dem Mischen hervorgehende Ausgabe-Farbsignal (Ao, Ro, Go, Bo) wird gemäß den Gleichungen (11)–(14) erzeugt. In den Gleichungen (11)–(14) stellen R1, G1, B1 und A1 jeweils die Rot-, Grün-, Blau- und Alpha-Komponenten des vernebelten und gefilterten Signals aus dem Block **310** dar. Rb, Gb, Bb und Ab stellen jeweils die Rot-, Grün-, Blau- und Alpha-Komponenten der Hintergrundfarbe dar. Für jede Farbsignalkomponente (Ao, Ro, Go, Bo) wird ein Ausgabe-Farbsignal erzeugt. Das Flussdiagramm endet am Block **314**.

[0052] Ist der Nebelfaktor gleich 0, dann wird kein Nebel angezeigt. In diesem Fall werden anschließend an das Filtern die gefilterten Texel mit einer Hintergrundfarbe gemischt, um ein Ausgabe-Farbsignal zu erzeugen. Das Ausgabe-Farbsignal (Ao, Ro, Go, Bo) wird gemäß den Gleichungen (6)–(9) erzeugt. In den Gleichungen (6)–(9) stellen R, G, B und A jeweils die Rot-, Grün-, Blau- und Alpha-Komponenten des gefilterten Signals aus dem Block **308** dar. Rb, Gb, Bb und Ab stellen jeweils die Rot-, Grün-, Blau- und Alpha-Komponenten der Hintergrundfarbe dar. Für jede Farbsignalkomponente wird ein Ausgabe-Farbsignal erzeugt.

[0053] Somit wurden ein Verfahren und eine Vorrichtung zum Farbstanzen gefilterter Texturen bereitgestellt. Obwohl die vorliegende Erfindung mit Bezugnahme auf spezifische Ausführungsbeispiele beschrieben wurde, ist es offensichtlich, dass verschiedene Modifikationen und Abänderungen an diesen Ausführungsformen vorgenommen werden können, ohne den Geltungsbereich der Erfindung zu verlassen, wie er in den Ansprüchen dargelegt ist. Dementsprechend sind die Beschreibung und die Zeichnungen eher in einem veranschaulichenden als in einem einschränkenden Sinne aufzufassen.

Patentansprüche

1. Ein Verfahren zum Farbstanzen (chroma keying) gefilterter Texturen in einem Computersystem, aufweisend die Schritte:
Bestimmen, ob ein Texel mit einer ersten Farbe mit einer vorgegebenen Farbe übereinstimmt, welche eine Transparenz anzeigt;
Ersetzen des Texels mit einer ersten Farbe durch ein Texel mit einer zweiten Farbe, wenn das Texel mit der ersten Farbe mit der vorgegebenen Farbe übereinstimmt;
Filtern der Texel, einschließlich der ersetzten Texel; und
Mischen der gefilterten Texel mit einer Hintergrundfarbe.

2. Das Verfahren nach Anspruch 1, wobei der Schritt des Ersetzens der Texel mit einer ersten Farbe vor dem Schritt des Filterns der Texel durchgeführt wird.

3. Das Verfahren nach Anspruch 1, wobei der Schritt des Filterns von Texeln den Schritt des Kantenglättens (antialiasing) von Texeln umfasst.

4. Das Verfahren nach Anspruch 1, wobei der Schritt des Filterns von Texeln den Schritt des bilinearen Filterns von Texeln umfasst.

5. Das Verfahren nach Anspruch 1, wobei der Schritt des Filterns von Texeln in einem gefilterten Farbsignal resultiert, wobei das gefilterte Farbsignal gefilterte Alpha-, Rot-, Grün- und Blau-Komponenten aufweist.

6. Das Verfahren nach Anspruch 1, wobei der Schritt des Filterns von Texeln in einem gefilterten Farbsignal resultiert, wobei das gefilterte Farbsignal gefilterte Komponenten des YCBCR-Farbmodells aufweist.

7. Das Verfahren nach Anspruch 1, wobei die zweite Farbe von der ersten Farbe verschieden ist.

8. Das Verfahren nach Anspruch 7, wobei das Texel mit einer zweiten Farbe ein Texel ist, welches Alpha-, Rot-, Grün- und Blau-Komponenten aufweist, die jeweils gleich 0 sind.

9. Das Verfahren nach Anspruch 7, wobei das Texel mit einer zweiten Farbe ein Texel des YCB-CR-Farbmodells ist, mit Alpha-, CR-, Y- und CB-Komponenten gleich 0 bzw. 128, 16 und 128.

10. Das Verfahren nach Anspruch 1, wobei dann, wenn das Texel mit einer ersten Farbe nicht mit der vorgegebenen Farbe übereinstimmt, die Alpha-Komponente des Farbsignals auf 1 gesetzt wird.

11. Das Verfahren nach Anspruch 5, wobei die Alpha-Komponenten (A) des gefilterten Farbsignals mit einer Alpha-Komponente(Ab) der Hintergrundfarbe gemischt wird, um ein Ausgabe-Farbsignal hervorzubringen, das im Wesentlichen auf der Formel $Ao = A + [(1 – A)\cdot Ab]$ basiert.

12. Das Verfahren nach Anspruch 5, wobei die Rot-Komponenten (R) des gefilterten Farbsignals mit einer Rot-Komponente (Rb) der Hintergrund gemischt wird, um ein Ausgabe-Farbsignal (Ro) hervorzubringen, welches im Wesentlichen auf der Formel $Ro = R + [(1 – A)\cdot Rb]$ basiert, wobei die Grün-Komponente (G) des gefilterten Farbsignals mit einer Grün-Komponente (Gb) der Hintergrundfarbe gemischt wird, um ein Ausgabe-Farbsignal (Go) hervorzubringen, welches im Wesentlichen auf der Formel $Go = G + [(1 – A)\cdot Gb]$ basiert, und wobei die Blau-Komponente (B) des gefilterten Farbsignals mit einer Blau-Komponente (Bb) der Hintergrundfarbe gemischt wird, um ein Ausgabe-Farbsignal (Bo) hervorzubringen, welches im Wesentlichen auf der Formel $Bo = B + [(1 – A)\cdot Bb]$ basiert.

13. Das Verfahren nach Anspruch 1, ferner mit dem Schritt des Vernebelns (fogging) von Texeln, wobei der Schritt des Vernebelns von Texeln gemäß einem Nebel-Faktor (f) durchgeführt wird.

14. Das Verfahren nach Anspruch 13, wobei der Schritt des Vernebelns vor dem Schritt des Mischens der gefilterten Texel mit einer Hintergrundfarbe durchgeführt wird.

15. Das Verfahren nach Anspruch 5, wobei die Rot-Komponente (R) des gefilterten Farbsignals mit einer Rot-Komponente eines Nebel-Signals (Rfog) kombiniert wird, um ein vernebeltes und gefiltertes Signal (R1) zu erzeugen, welches im Wesentlichen auf der Formel $R1 = [R\cdot(1 – f)] + (Rfog\cdot f \cdot A)$ basiert, wobei die Grün-Komponente (G) des gefilterten Farbsignals mit einer Grün-Komponente eines Nebel-Signals (Gfog) zur Hervorbringung eines vernebelten und gefilterten Signals (G1) kombiniert wird, welches im Wesentlichen auf der Formel $G1 = [G\cdot(1 – f)] + (Gfog\cdot f\cdot A)$ basiert, und wobei die Blau-Komponente (B) des gefilterten Farbsignals mit einer Blau-Komponente einer Nebel-Farbe (Gfog) zur Hervorbringung eines vernebelten und gefilterten Signals (B1) kombiniert wird, welches im Wesentlichen auf der Formel $B1 = [B\cdot(1 – f)] + (Bfog\cdot f\cdot A)$ basiert.

16. Das Verfahren nach Anspruch 15, wobei das vernebelte und gefilterte Signal (R1) mit der Rot-Komponente der Hintergrundfarbe (Rb) gemischt wird, um ein Ausgabe-Signal (Ro) hervorzubringen, welches im Wesentlichen auf der Formel $Ro = R1 + [(1 – A1)\cdot Rb]$ basiert, wobei das vernebelte und gefilterte Signal (G1) mit der Grün-Komponente der Hintergrundfarbe (Gb) gemischt wird, um ein Ausgabe-Farbsignal (Go) hervorzubringen, welches im Wesentlichen auf der Formel $Go = G1 + [(1 – A1)\cdot Gb]$ basiert, und wobei das vernebelte und gefilterte Signal (B1) mit der Blau-Komponente der Hintergrundfarbe (Bb) gemischt wird, um ein Ausgabe-Farbsignal (Bo) hervorzubringen, welches im Wesentlichen auf der Formel $Bo = B1 + [(1 – A1)\cdot Bb]$ basiert.

17. Ein Computersystem, aufweisend:
einen Speicher;
einen Prozessor-Subsystem, welches mit dem Speicher gekoppelt ist, wobei das Prozessor-Subsystem bestimmt, ob ein Texel mit einer ersten Farbe mit einer vorgegebenen Farbe übereinstimmt, welche eine Transparenz anzeigt, das Texel mit einer ersten Farbe mit einem Texel einer zweiten Farbe ersetzt, wenn das Texel mit der ersten Farbe mit der vorgegebenen Farbe übereinstimmt, die Texel filtert und die gefilterten Texel mit einer Hintergrundfarbe mischt.

18. Das Computersystem nach Anspruch 17, wobei das Prozessor-Subsystem einen Prozessor und einen Co-Prozessor aufweist.

19. Das Computersystem nach Anspruch 17, wobei der Prozessor eine Pixel-Engine aufweist, die mit dem Prozessor-Subsystem gekoppelt ist, wobei die Pixel-Engine eine Schaltung zum Bestimmen enthält, ob ein Texel mit einer ersten Farbe mit einer vorgegebenen Farbe übereinstimmt, wobei die Pixel-Engine-Schaltung das Texel mit einer ersten Farbe ersetzt, wenn das Texel mit der ersten Farbe mit der vorgegebenen Farbe übereinstimmt.

20. Das Computersystem nach Anspruch 19, wobei die Pixel-Engine eine Schaltung zum Filtern von Texeln enthält und zum Mischen der gefilterten Texel mit einer Hintergrundfarbe.

21. Das Computersystem nach Anspruch 17, wobei das Prozessor-Subsystem die Texel durch Antia-

liasing filtert.

22. Das Computersystem nach Anspruch 17, wobei das Prozessor-Subsystem die Texel durch bilineare Filterung filtert.

23. Das Computersystem nach Anspruch 17, wobei die zweite Farbe von der ersten Farbe verschieden ist.

24. Das Computersystem nach Anspruch 23, wobei das Texel einer zweiten Farbe ein Texel mit Alpha-, Rot-, Grün- und Blau-Komponenten mit jeweils einem Wert von 0 ist.

25. Das Computersystem nach Anspruch 23, wobei das Texel einer zweiten Farbe ein Texel mit Alpha-, CR-, Y- und CB-Komponenten gleich 0, 128, 16 bzw. 128 ist.

26. Das Computersystem nach Anspruch 17, wobei dann, wenn das Texel mit der ersten Farbe nicht mit der vorgegebenen Farbe übereinstimmt, das Prozessor-Subsystem die Farbsignal-Alpha-Komponente auf 1 setzt.

27. Das Computersystem nach Anspruch 17, wobei die Alpha-Komponente (A) des gefilterten Farbsignals mit einer Alpha-Komponente (Ab) der Hintergrundfarbe gemischt wird, zum Hervorbringen eines Ausgabe-Farbsignals, welches im Wesentlichen auf der Formel A0 = A + [(1 – A)·Ab] basiert.

28. Das Computersystem nach Anspruch 17, wobei die Rot-Komponente (R) des gefilterten Farbsignals mit einer Rot-Komponente (Rb) der Hintergrundfarbe gemischt wird, um ein Ausgabe-Farbsignal (Ro) bereitzustellen, welches im Wesentlichen auf der Formel Ro = R + [(1 – A)·Rb] basiert, wobei die Grün-Komponente (G) des gefilterten Farbsignals mit einer Grün-Komponente (Gb) der Hintergrundfarbe zum Hervorbringen eines Ausgabe-Farbsignals (Go) gemischt wird, welches im Wesentlichen auf der Formel Go = G + [(1 – A)·Gb] basiert, und wobei die Blau-Komponente (B) des gefilterten Farbsignals mit einer Blau-Komponente (Bb) der Hintergrundfarbe zum Hervorbringen eines Ausgabe-Farbsignals (Bo) gemischt wird, welches im Wesentlichen auf der Formel Bo = B + [(1 – A)·Bb] basiert.

29. Das Computersystem nach Anspruch 20, wobei die Pixel-Engine eine Vernebelungs(fogging)-Schaltung enthält.

30. Das Computersystem nach Anspruch 29, wobei die Farbkomponenten vor der Mischung der gefilterten Texel mit einer Hintergrundfarbe vernebelt werden.

31. Das Computersystem nach Anspruch 17, wobei die Rot-Komponente (R) des gefilterten Farbsignals mit einer Rot-Komponente eines Nebel-Signals (Rfog) kombiniert wird, um ein vernebeltes und gefiltertes Signal (R1) bereitzustellen, welches im Wesentlichen auf der Formel R1 = [R·(1 – f)] + (Rfog·f·A) basiert, wobei die Grün-Komponente (G) des gefilterten Farbsignals mit einer Grün-Komponente eines Nebel-Signals (Gfog) kombiniert wird, um ein vernebeltes und gefiltertes Signal (G1) bereitzustellen, welches im Wesentlichen auf der Formel G1 = [G·(1 – f)] + (Gfog·f·A) basiert, und wobei die Blau-Komponente (B) des gefilterten Farbsignals mit einer Blau-Komponente einer Nebel-Farbe (Gfog) kombiniert wird, um ein vernebeltes und gefiltertes Signal (B1) bereitzustellen, welches im Wesentlichen auf der Formel B1 = [B·(1 – f)] + (Bfog·f·A) basiert.

32. Das Computersystem nach Anspruch 31, wobei das vernebelte und gefilterte Signal (R1) mit der Rot-Komponente der Hintergrundfarbe (Rb) gemischt wird, um ein Ausgabe-Farbsignal (Ro) bereitzustellen, welches im Wesentlichen auf der Formel Ro = R1 + [(1 – A1)·Rb] basiert, wobei das vernebelte und gefilterte Signal (G1) mit der Grün-Komponente der Hintergrundfarbe (Gb) gemischt wird, um ein Ausgabe-Farbsignal (Go) bereitzustellen, welches im Wesentlichen auf der Formel Go = G1 + [(1 – A1)·Gb] basiert, und wobei das vernebelte und gefilterte Signal (B1) mit der Blau-Komponente der Hintergrundfarbe (Bb) gemischt wird, um ein Ausgabe-Farbsignal (Bo) bereitzustellen, welches im Wesentlichen auf der Formel Bo = B1 + [(1 – A1)·Bb] basiert.

33. Das Verfahren nach Anspruch 1, wobei der Schritt des Filterns von Texeln in einem gefilterten Farbsignal resultiert, wobei das gefilterte Farbsignal n-Farbkomponenten aufweist.

34. Das Verfahren nach Anspruch 33, wobei eine Farbkomponente (Xn) des gefilterten Farbsignals mit einer entsprechenden Farbkomponente (Xnb) einer Hintergrundfarbe gemischt wird, um ein Ausgabe-Farbsignal (Xno) bereitzustellen, welches im Wesentlichen auf der Formel Xno = Xn + [(1 – A)·Xnb] basiert.

35. Das Verfahren nach Anspruch 33, wobei die Farbkomponente (Xn) des gefilterten Farbsignals mit einer entsprechenden Farbkomponente eines Nebelsignals (Xnfog) kombiniert wird, um ein vernebeltes und gefiltertes Signal (Xn1) bereitzustellen, welches im Wesentlichen auf der Formel Xn1 = [Xn·(1 – f)] + (Xnfog·f·A) basiert.

36. Das Verfahren nach Anspruch 35, wobei das vernebelte und gefilterte Signal (Xn1) mit einer entsprechenden Farbkomponente der Hintergrundfarbe (Xnb) gemischt wird, um ein Ausgabe-Farbsignal (Xno) bereitzustellen, welches im Wesentlichen auf der Formel Xno = Xn1 + [(1 – A1)·Xnb] basiert.

37. Das Computersystem nach Anspruch 17, wobei eine Farbkomponente (Xn) des gefilterten Farbsignals mit einer entsprechenden Farbkomponente (Xnb) einer Hintergrundfarbe gemischt wird, um ein Ausgabe-Farbsignal (Xno) bereitzustellen, welches im Wesentlichen auf der Formel Xno = Xn + [(1 – A)·Xnb] basiert.

38. Das Computersystem nach Anspruch 17, wobei die Farbkomponente (Xn) des gefilterten Farbsignals mit einer entsprechenden Farbkomponente eines Nebel-Signals (Xnfog) kombiniert wird, um ein vernebeltes und gefiltertes Signal (Xn1) bereitzustellen, welches im Wesentlichen auf der Formel Xn1 = [Xn·(1 – f)] + (Xnfog·f·A) basiert.

39. Das Computersystem nach Anspruch 38, wobei das vernebelte und gefilterte Signal (Xn1) mit einer entsprechenden Farbkomponente der Hintergrundfarbe (Xnb) gemischt wird, um ein Ausgabe-Farbsignal (Xno) bereitzustellen, welches im Wesentlichen auf der Formel Xno = Xn1 + [(1 – A1)·Xnb] basiert.

40. Eine Schaltung zum Farbstanzen (chroma keying) von gefilterten Texturen, aufweisend:
Logik zum Bestimmen, ob ein Texel mit einer ersten Farbe mit einer vorgegebenen Farbe übereinstimmt, welche eine Transparenz anzeigt;
Logik zum Ersetzen des Texels mit einer ersten Farbe durch ein Texel mit einer zweiten Farbe, wenn das Texel mit einer ersten Farbe mit der vorgegebenen Farbe übereinstimmt;
Logik zum Filtern von Texeln einschließlich der ersetzten Texel;
Logik zum optionalen Durchführen von Vernebelungs(fogging)-Operationen an den gefilterten Texeln;
Logik zum Mischen der gefilterten Texel mit einer Hintergrundfarbe.

Es folgen 5 Blatt Zeichnungen

Anhängende Zeichnungen

FIG. 1

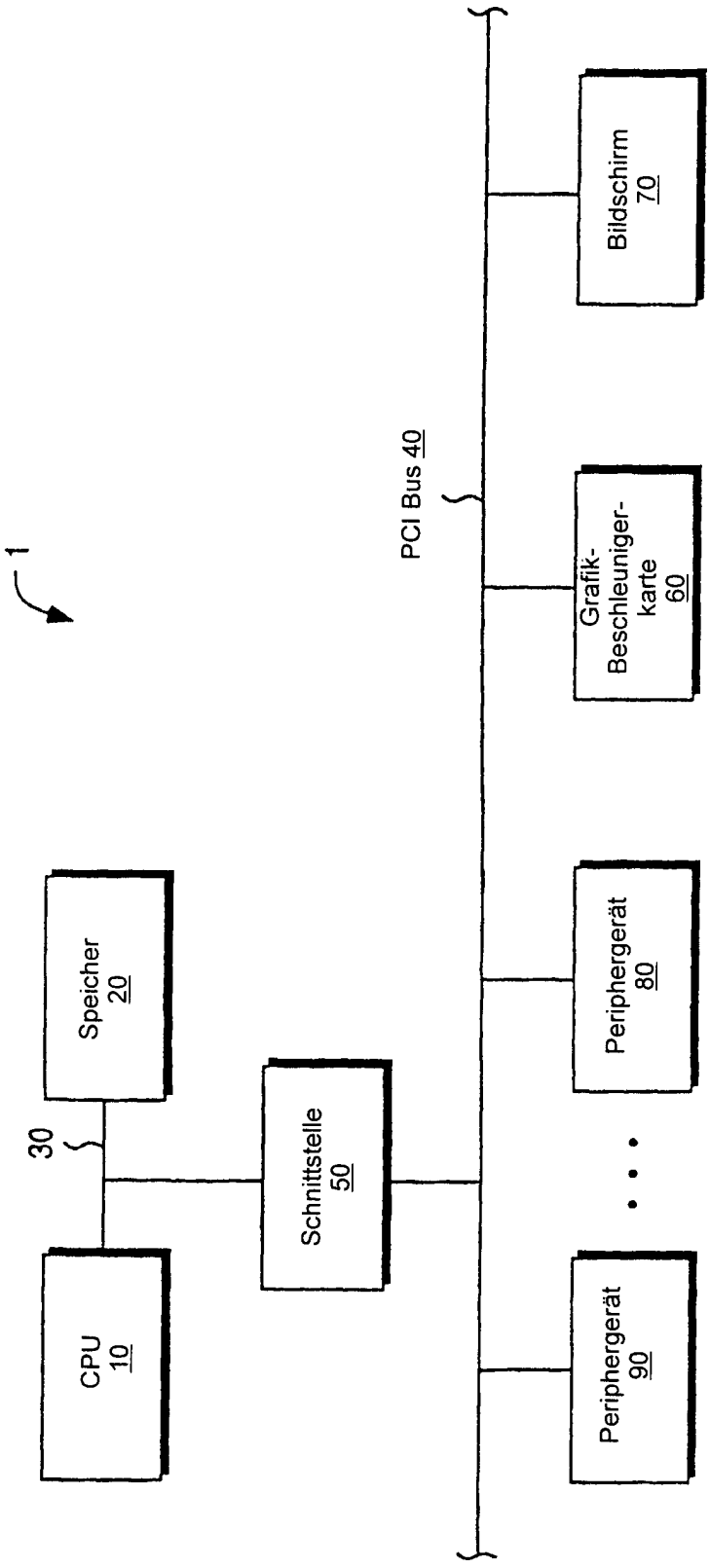
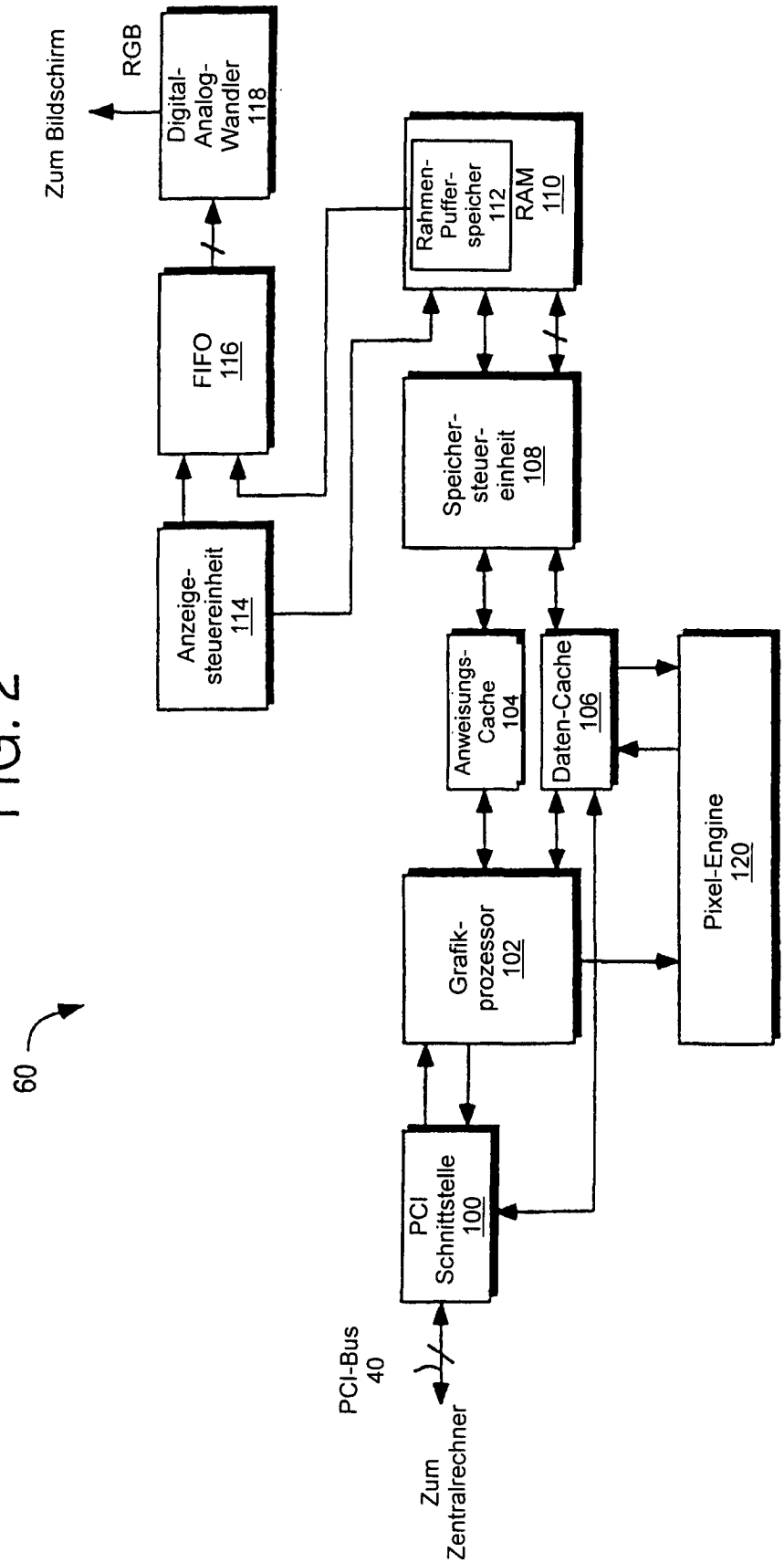


FIG. 2


60
Zum Bildschirm
RGB
Digital-Analog-Wandler 118
FIFO 116
Anzeige-steuereinheit 114
Rahmen-Puffer-speicher 112
RAM 110
Speicher-steuer-einheit 108
Anweisungs-Cache 104
Daten-Cache 106
Grafik-prozessor 102
Pixel-Engine 120
PCI Schnittstelle 100
PCI-Bus 40
Zum Zentralrechner

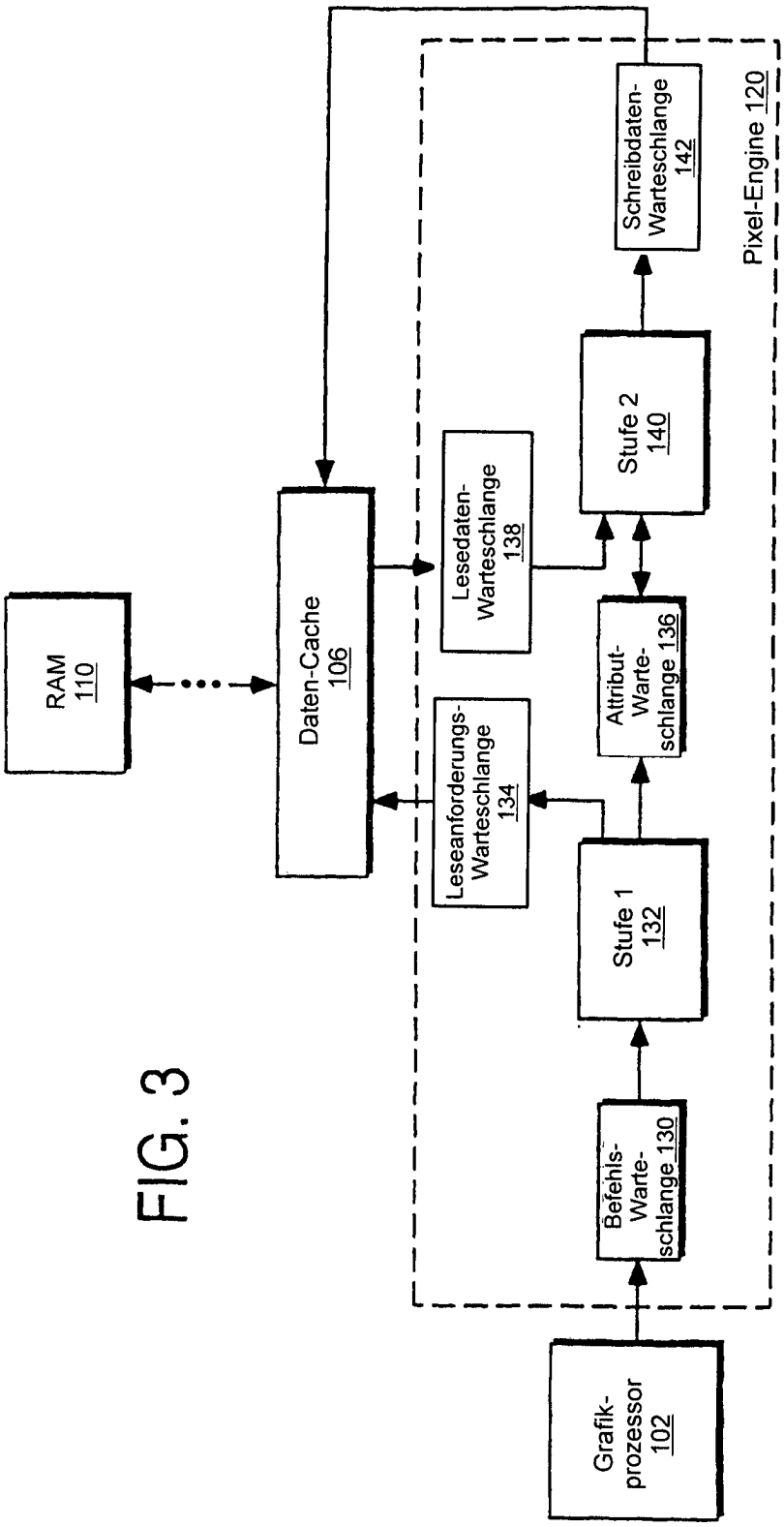

RAM
110
Daten-Cache
106
Leseanforderungs-
Warteschlange
134
Lesedaten-
Warteschlange
138
Befehls-
Warte-
schlange 130
Attribut-
Warte-
schlange 136
Schreibdaten-
Warteschlange
142
Stufe 1
132
Stufe 2
140
Grafik-
prozessor
102
Pixel-Engine 120

FIG. 3

FIG. 4

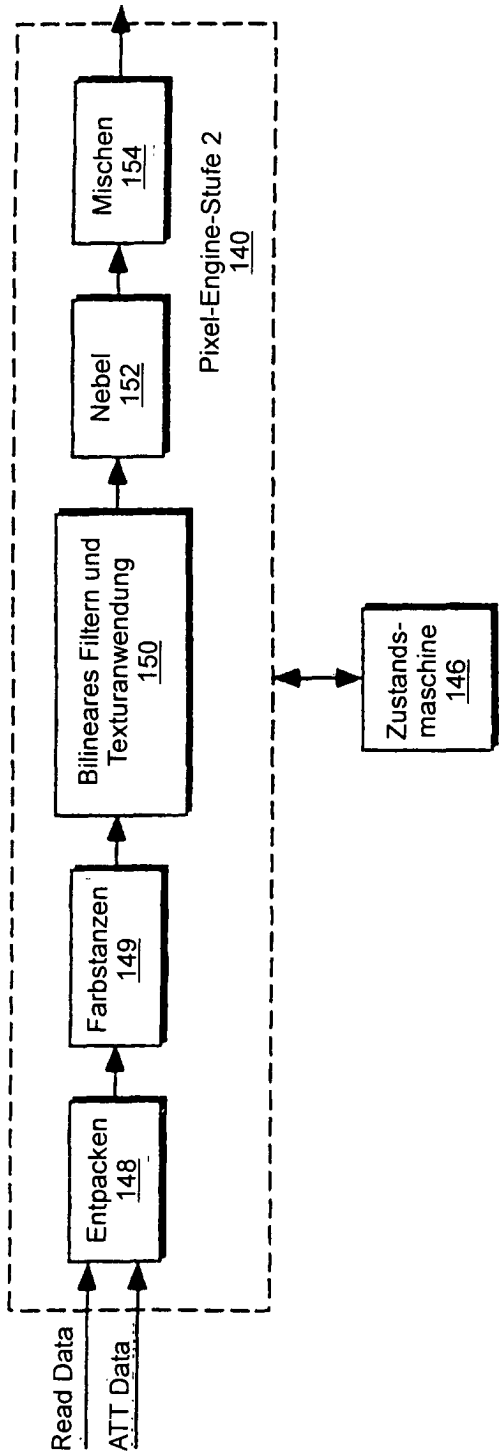
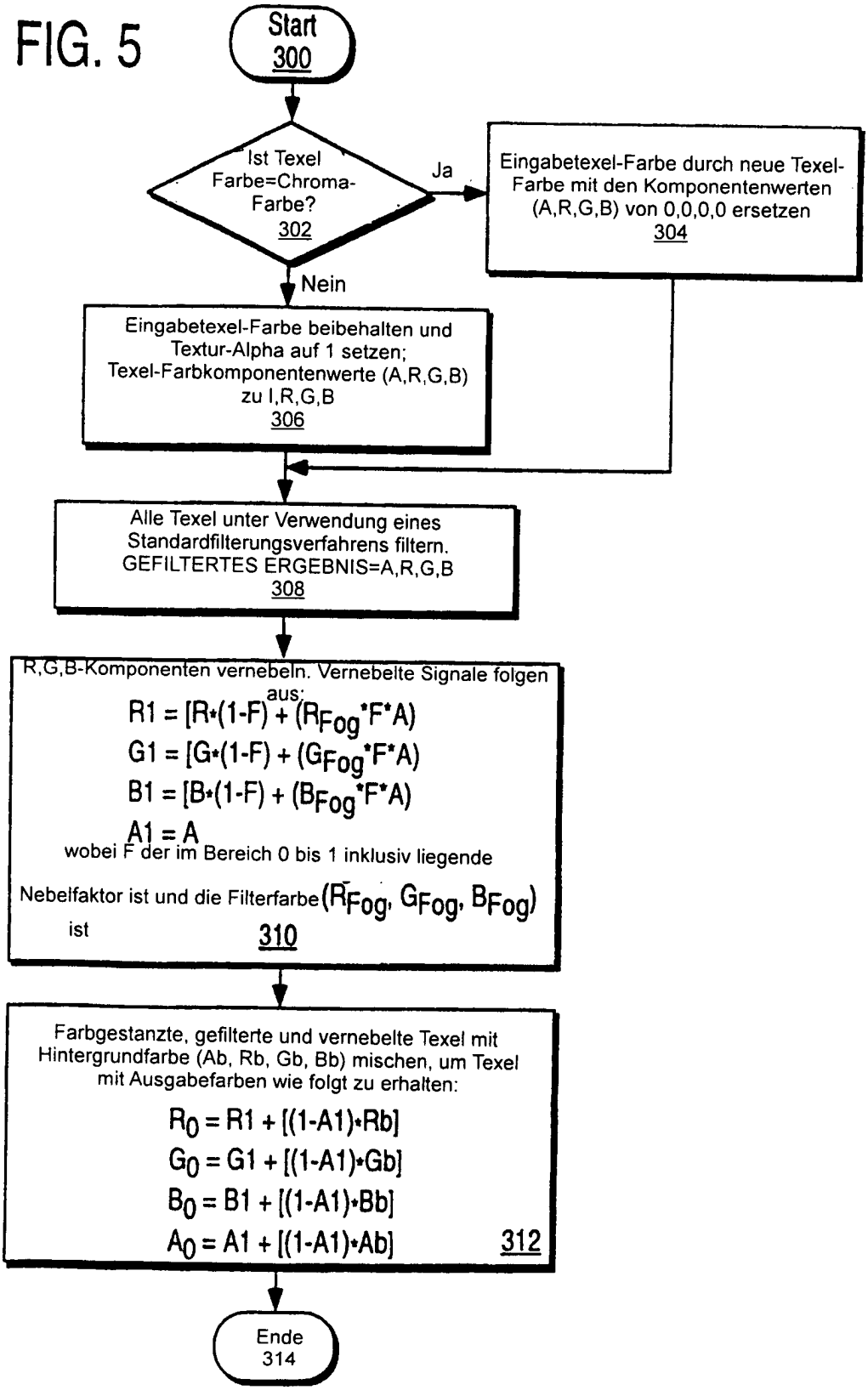

Read Data
ATT Data
Entpacken 148
Farbstanzen 149
Bilineares Filtern und Texturanwendung 150
Nebel 152
Mischen 154
Pixel-Engine-Stufe 2 140
Zustands-maschine 146

FIG. 5


Start
300
Ist Texel Farbe=Chroma-Farbe?
302
Ja
Eingabetexel-Farbe durch neue Texel-Farbe mit den Komponentenwerten (A,R,G,B) von 0,0,0,0 ersetzen
304
Nein
Eingabetexel-Farbe beibehalten und Textur-Alpha auf 1 setzen; Texel-Farbkomponentenwerte (A,R,G,B) zu I,R,G,B
306
Alle Texel unter Verwendung eines Standardfilterungsverfahrens filtern. GEFILTERTES ERGEBNIS=A,R,G,B
308
R,G,B-Komponenten vernebeln. Vernebelte Signale folgen aus:
R1 = [R*(1-F) + (RFog*F*A)
G1 = [G*(1-F) + (GFog*F*A)
B1 = [B*(1-F) + (BFog*F*A)
A1 = A
wobei F der im Bereich 0 bis 1 inklusiv liegende Nebelfaktor ist und die Filterfarbe (RFog, GFog, BFog) ist
310
Farbgestanzte, gefilterte und vernebelte Texel mit Hintergrundfarbe (Ab, Rb, Gb, Bb) mischen, um Texel mit Ausgabefarben wie folgt zu erhalten:
R0 = R1 + [(1-A1)*Rb]
G0 = G1 + [(1-A1)*Gb]
B0 = B1 + [(1-A1)*Bb]
A0 = A1 + [(1-A1)*Ab]
312
Ende
314