



(19)
Bundesrepublik Deutschland
Deutsches Patent- und Markenamt

(10) DE 600 24 514 T2 2006.07.27

(12)

Übersetzung der europäischen Patentschrift

(97) EP 1 031 946 B1

(21) Deutsches Aktenzeichen: 600 24 514.4

(96) Europäisches Aktenzeichen: 00 301 217.6

(96) Europäischer Anmeldetag: 16.02.2000

(97) Erstveröffentlichung durch das EPA: 30.08.2000

(97) Veröffentlichungstag
der Patenterteilung beim EPA: 07.12.2005

(47) Veröffentlichungstag im Patentblatt: 27.07.2006

(51) Int Cl.⁸: G06T 15/70 (2006.01)
G06T 17/40 (2006.01)

(30) Unionspriorität:
4553599 23.02.1999 JP

(84) Benannte Vertragsstaaten:
DE, ES, FR, GB, IT

(73) Patentinhaber:
Sega Corp., Tokio/Tokyo, JP

(72) Erfinder:
Saito, Tomoaki, Tokyo, JP; Ando, Takashi, Tokyo, JP

(74) Vertreter:
**Benedum, U., Dipl.-Chem.Univ.Dr.rer.nat.,
Pat.-Anw., 80333 München**

(54) Bezeichnung: **Aufzeichnungsmedium, Verfahren und Einrichtung zur Bildverarbeitung mit integrierten Daten
zur Formmodellierung**

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 eingelebt, 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**HINTERGRUND DER ERFINDUNG****1. Gebiet der Erfindung**

[0001] Die Erfindung betrifft ein Bildverarbeitungsverfahren zum Verarbeiten von Bildern in Echtzeit, ein Bildverarbeitungsverfahren zum Erzeugen einer Bilddatenstruktur, die sich zum Verarbeiten von Bildern in Echtzeit eignet, und ein Aufzeichnungsmedium, das die betreffende Bilddatenstruktur bzw. das betreffende Bildverarbeitungsprogramm aufzeichnet.

2. Beschreibung des Stands der Technik

[0002] In einer Maschine für Heimvideospiele werden bewegte Bilder von Objekten eines Spiels in Echtzeit synchron zum Spielfortschritt erzeugt. Die Bildverarbeitung in einer solchen Spielemaschine muss die Positionen der Objekte im Spiel abhängig von den eingegebenen Steuersignalen des Benutzers in einer kurzen Zeitspanne bewegen, beispielsweise in einer Frameperiode, damit Bilder gezeichnet werden, die den Bewegungspositionen im Bildspeicher (Framespeicher) entsprechen.

[0003] Damit die Bewegungen natürlicher ablaufen, ist es erforderlich, dass die Figuren, d. h. die Lebewesen, als Objekte des Spiels Gelenkteile und Muskelteile zusammen mit der Bewegung der Gliedmaßen und Hälse bewegen können, und dass die Gelenkteile abhängig von der Bewegung der anderen Polygone eine unterschiedliche Form annehmen. Anders ausgedrückt müssen für ein natürlicheres Aussehen der Bewegung des Teils zwischen dem Oberarm und der Schulter einer Person oder des Teils zwischen dem Oberarm und dem Unterarm die Muskeln des Oberarms und der Schulter abhängig von der Bewegung der Polygone anschwellen.

[0004] Ein Zeichnungsverfahren, das der Bewegung solcher Gelenkteile ein natürliches Aussehen verleiht, spezifiziert die Ecken, die für Modelle zu beiden Seiten des Gelenks verbunden werden müssen, und fügt für das Gelenk zwischen den Ecken ein Polygon zu. Mit einem derartigen Zeichnungsverfahren wird das Gelenk jedoch mit standardisierten Polygone gezeichnet. Dadurch kann man keine Bewegung zum Ausdruck bringen, die der Winkeländerung entspricht, und die erzeugten Bilder erscheinen dadurch unnatürlich.

[0005] Um die Bewegung natürlicher zu zeichnen, wird vorgeschlagen, integrierte Formgebungsmodelle zu verwenden, die als "Hüllen" bezeichnet werden. Dabei werden auf dem Gebiet der Bildverarbeitung für Spielfilme, beispielsweise CG-Spielfilme (Filme, in denen Computergraphik verwendet wird) dreidimensionale Modellierer verwendet, für die keine

Echtzeitverarbeitung erforderlich ist. In diesen integrierten Formgebungsmodellen bestehen die Objekte (Figuren) aus Polygonen und Knochen, die die Polygone beeinflussen. Anders ausgedrückt besteht ein integriertes Formgebungsmodell aus Polygonen, die die Außenfläche der Figur bilden und abhängig von der momentanen Bewegung gezeichnet werden, und aus Knochen, die die Positionen der Polygone mit einer vorbestimmten Gewichtung beeinflussen. Neue Positionen von Knochen, die die Positionen der Polygone beeinflussen, werden mit Hilfe einer Positionsumsetzmatrix abhängig von der Operation berechnet, die der Benutzer eingibt. Die Positionen der Ecken der Polygone werden anhand der Größe des Einflusses (Gewichtung) der Knochen ermittelt. Komplizierte Bewegungen kann man durch Polygone ausdrücken, indem man die Polygone abhängig von den ermittelten Ecken rendert (zeichnet).

[0006] Die Zeichnung in [Fig. 1](#) zeigt ein Beispiel für die Bewegung des genannten integrierten Formgebungsmodells in einem CG-Spielfilm. Das in [Fig. 1](#) dargestellte Modell besteht aus einem Armteil **1** und einem Handteil **2**. [Fig. 1A](#) zeigt die Grundform des Modells. Wird der Arm gebogen, so dass bezogen auf die Grundform der Oberarm **3** des Armtyps **1** stärker senkrecht verläuft und der Unterarmteil **4** des Armtyps **1** stärker waagrecht verläuft, so erwartet man den in [Fig. 1B](#) dargestellten Status. Anders ausgedrückt zeigt das Gelenk, das den Oberarm **3** und den Unterarmteil **4** verbindet, eine natürlich gebogene Form, und die inneren Muskeln **3A** des Oberarms **3** treten hervor.

[0007] [Fig. 2](#) zeigt eine erläuternde Zeichnung, die eine herkömmliche Bildverarbeitung mit Hilfe eines integrierten Formgebungsmodells in CG-Spielfilmen darstellt. [Fig. 3](#) zeigt ein Flussdiagramm der zugehörigen Bildverarbeitungsprozedur. [Fig. 2](#) zeigt die Datenstruktur **620** des integrierten Formgebungsmodells (Hülle). In diesem Beispiel befinden sich insgesamt sieben Modelle, die Modelle 1–7, in der baumartigen hierarchischen Struktur in [Fig. 2](#). Die Datenstruktur eines jeden Modells gleicht den Daten **624** des Modells 4 und den Daten **627** des Modells 7.

[0008] Besitzt ein Modell darzustellende Polygone, so weist das Modell Ecken auf, die die Bausteine des Polygons bilden. Daher enthält die Datenstruktur **624** des Modells 4 die Eckenliste **632** und die Polygonliste **635** der Polygone, die aus einer Verknüpfung der Ecken bestehen. Die Eckenliste **632** enthält Positionsinformationen und Normalendaten einer jeden Ecke in den lokalen Koordinaten des Modells. Jedes Modell weist auch eine Modellmatrix **631** auf, die Information über die relative Position bezüglich eines Modells in der höheren Hierarchie enthält. Das Modell 4 besitzt beispielsweise Drehung, Verschiebung (Translation) und Größe (Skalierung) als relative Positionsinformation bezüglich des Modells 3. Die aus

derartigen Positionsinformationen bestehende Modellmatrix **631** kann auch Matrizedaten enthalten, wenn das Koordinatensystem des Modells 3 in das Koordinatensystem des Modells 4 umgewandelt wird.

[0009] Die Datenstruktur **624** des Modells 4 weist ebenfalls eine Gewichtungsliste **633** auf, die die Ecken weiterer Modelle enthält, die das Modell 4 beeinflusst, und die Gewichtungswerte, die die Stärke dieses Einflusses angeben. Für das Beispiel in **Fig. 1** beeinflusst beispielsweise das Modell des Unterarms **4** den Muskelabschnitt **3A** des Oberarms **3** mit einem vorbestimmten Gewichtungswert. Daher sind in der Gewichtungsliste des Modells für den Unterarmabschnitt **4** die Kennung (ID) des Modells des Oberarms **3**, die Indizes der zugehörigen Ecken (Eckenkennungen) und die Gewichtungswerte enthalten.

[0010] Ist das Modell 4 der Modellstruktur in **Fig. 2** der Unterarmabschnitt **4** in **Fig. 1** und das Modell 7 der Oberarmteil **3**, so verändern sich die Positionen der Ecken des Muskelteils **3A** im Modell des Oberarmteils **3** und die Bewegung des Unterarmabschnitts **4**, und es wird das Bild mit dem hervortretenden Muskelteil **3A** gezeichnet. Nun wird die Bildverarbeitungsprozedur durch einen dreidimensionalen Modellierer, der für CG-Filme generell verwendet wird, anhand von **Fig. 3** erklärt. Zuerst wird die Matrix, die die Bewegungen aller Modelle implementiert, berechnet und in der Datenstruktur eines jeden Modells gespeichert (Schritt **602**). Nun werden unter Bezug auf die Gewichtungsliste **633** die Eckenkoordinaten und die Normalendaten (Eckendaten), die die Eckenindizes (Eckenkennungen) des Modells (Modell 7) bezeichnen, die vom Zielmodell (Modell 4) beeinflusst werden, aus der Eckenliste **632** des Modells 7 gelesen (Schritt **603**). Die gelesenen Eckendaten werden in Eckendaten konvertiert, falls die Eckendaten zum lokalen Koordinatensystem des Modells 4 gehören, und zwar abhängig vom Zusammenhang in der Grundform. Die konvertierten Eckenkoordinaten und Normalendaten werden mit der Modellmatrix des Modells 4 multipliziert (Schritt **604**). Die mit der Modellmatrix multiplizierten Eckenkoordinaten und Normalendaten werden mit dem Gewichtungswert in der Gewichtungsliste **633** multipliziert, und die Werte werden mit den Werten in der Gewichtungseckenliste **634** in der Datenstruktur **627** des Modells 7 vereint.

[0011] Die obigen Prozeduren **603**, **604** und **605** werden für die gesamte Gewichtungsliste ausgeführt (Schritt **606**), und sie werden für alle Modelle ausgeführt (Schritt **607**). Dadurch werden Koordinatenumsetzungen ausgeführt, die der Bewegung der Figuren an den tatsächlichen Ecken aller Modelle entsprechen, sowie die Positionsbewegung abhängig vom Grad des Einflusses durch andere Modelle. Nun wer-

den die Normalenvektoren der Gewichtungseckenliste aller Modelle normiert, so dass man die Normalenvektoren zur Lichtquellenverarbeitung beim Rendern verwenden kann, das später erfolgt (Schritt **608**). Anhand der Eckenpositionen in der Gewichtungseckenliste wird das Zeichnen (Rendern) der Polygone in der Polygonliste vorgenommen (Schritt **609**).

[0012] Für die obigen Operationen werden die Matrixoperationen und Gewichtungsoperationen für die Ecken aller Modelle ausgeführt. Die Renderungsverarbeitung für die Polygone erfolgt nach dem Abschluss der obigen Operationen. Eine andere Möglichkeit ist, sich auf eine Ecke zu konzentrieren und die Matrixoperation und die Gewichtungsoperation der Modelle auszuführen, die die Ecke beeinflussen, anschließend die neue Eckenkoordinate der Ecke zu bestimmen und schließlich das Rendern auszuführen.

[0013] Dieses Bildverarbeitungsverfahren für CG-Spielfilme lässt sich jedoch nicht auf die Hochgeschwindigkeits-Bildverarbeitung übertragen, die in Echtzeit in einer Spieldatenmaschine stattfindet. Werden die Matrixoperation und die Gewichtungsoperation für alle Ecken aller Modelle ausgeführt, so müssen die Operationen anhand der Eckenlisten und Modellmatrizen unterschiedlicher Modelle in den Datenstrukturen des Modells vorgenommen werden. Werden derartige Operationen hinsichtlich einer Referenzprozedur ausgeführt, wobei ein Zeiger in der hierarchischen Struktur der im Speicher verteilten Modelldaten verwendet wird, so muss der Zeiger häufig verändert werden. Es gibt Einschränkungen beim Ausführen solcher Operationen in einer Spieldatenmaschine, da die Arbeitseffizienz des in der CPU der Spieldatenmaschine untergebrachten Cachespeichers gering ist. Werden die Matrixoperation und die Gewichtungsoperation für eine Ecke unter Konzentration auf diese Ecke vorgenommen, so müssen die Pufferbereiche für die Gewichtungsoperation für alle Ecken gespeichert werden. Wird diese Operation in einer einfach strukturierten Spieldatenmaschine vorgenommen und nicht in einem Supercomputer, der eine sehr große Speicherkapazität hat und äußerst schnell arbeitet, so ist der erforderliche Speicherumfang enorm, und eine Echtzeitverarbeitung ist schwierig. Deshalb kann man die Operation so nicht ausführen.

[0014] "Animals and Anatomy", Wilhelms J., IEEE Computer Graphics and Applications, IEEE INC. NEW YORK, vol. 17, no. 3, pages 22–30, 1997, offenbart das Modellieren der Gliederbewegung von Lebewesen, die ein Innenskelett, Muskeln und Haut besitzen. Die Bewegung der Haut wird dadurch modelliert, dass man Hautpunkte auf einem Hautgitter zusammen mit Ankerpunkten auf den darunter liegenden Knochen- und Muskelstrukturen betrachtet.

[0015] CA 2201466 offenbart das Erzeugen einer Skelethierarchie zum Einsatz in der Modellanimation. Die Skelethierarchie kann bewegte Kettenelemente und Nicht-Kettenelemente enthalten.

ZUSAMMENFASSUNG DER ERFINDUNG

[0016] Hinsichtlich des Gesagten ist es eine Aufgabe der Erfindung, ein Bildverarbeitungsverfahren bereitzustellen, das die Bildverarbeitung mit integrierten Formgebungsmodellen in Echtzeit erlaubt, sowie ein Bildverarbeitungsverfahren zum Erzeugen von Bilddaten auf integrierten Formgebungsmodellen, die für die Bildverarbeitung des Modells verwendet werden, sowie ein Aufzeichnungsmedium, das die Bilddaten und das verwendete Bildverarbeitungsprogramm aufzeichnet.

[0017] Es ist eine weitere Aufgabe der Erfindung, ein Bildverarbeitungsverfahren bereitzustellen, das auf einem Konverterprogramm zum Erzeugen von Bilddaten beruht und die Bildverarbeitung mit Hilfe integrierter Formgebungsmodelle in Echtzeit erlaubt, und zwar für dreidimensionale Modelldaten.

[0018] Es ist noch eine weitere Aufgabe der Erfindung, ein Bildverarbeitungsverfahren bereitzustellen, das Gelenkteile entsprechend der Bewegung einer Figur in eine natürliche Form transformiert, sowie ein Aufzeichnungsmedium, das ein Bildverarbeitungsprogramm zum Ausführen des Bildverarbeitungsverfahrens aufzeichnet, und ein Aufzeichnungsmedium, das Daten aufzeichnet, die eine für die Bildverarbeitung geeignete Struktur aufweisen.

[0019] Ein erster Aspekt der Erfindung stellt ein computerlesbares Aufzeichnungsmedium bereit, das integrierte Formgebungs-Modelldaten aufzeichnet, geeignet zum Modellieren der Gliederbewegung eines Objekts, wobei zahlreiche Modelle über eine hierarchische Struktur verknüpft sind, und mindestens ein erstes Modell zahlreiche Ecken aufweist, die Polygone bilden, und zumindest die Position einer ersten Ecke von den Positionen von zwei oder mehr der zahlreichen Modelle und von Gewichtungswerten aus diesen zwei oder mehr Modellen beeinflusst wird, und die integrierten Formgebungs-Modelldaten umfassen:

Formatdaten eines gemeinsamen Eckenpuffers, der Daten über die zahlreichen Ecken in den zahlreichen Modellen speichert, und zwar für jedes Modell in der hierarchischen Struktur;

eine Eckenliste, die für jedes der Modelle erzeugt wird, das die Ecken beeinflusst, und in der Eckendaten über eine Eckenkennung im gemeinsamen Eckenpuffer festgelegt sind;

Modellmatrixdaten, die für jedes der Modelle erzeugt werden und die Modellpositionen der zahlreichen Modelle in der Ursprungsform einstellen; und

eine Polygonliste, die für jedes der Modelle erzeugt

wird, das die Polygone aufweist, und die Polygondaten enthält, bei denen die Eckenkennung zugeordnete Daten sind, wobei die Eckendaten in jeder Eckenliste umfassen: mindestens die Positionsdaten einer Ecke, einen Gewichtungswert aus dem Modell, zu dem die Eckenliste gehört, und eine Eckenkennung im gemeinsamen Eckenpuffer, die der Ecke zugeordnet ist, und wobei Eckendaten der ersten Ecke, die von den Positionen der zwei oder mehr der zahlreichen Modelle beeinflusst werden, auf die Eckenlisten der zwei oder mehr Modelle als verteilte Eckendaten aufgeteilt sind.

[0020] Das obige Aufzeichnungsmedium, das die integrierten Formgebungsmodelldaten aufzeichnet, erlaubt die Bildverarbeitung, die zum Echtzeit-Zeichnen von Modellen in einer Spieldatenmaschine erforderlich ist.

[0021] In einer Ausführungsform des obigen Aufzeichnungsmediums wird jede der Eckenlisten, die verteilte Eckendaten enthält, in Anfangsgewichtsecken unterteilt, für die eine Eckenoperation zuerst ausgeführt wird, wenn die hierarchische Struktur der Modelle durchlaufen wird, Mittelgewichtsecken, für die die Eckenoperation nicht zuerst oder zuletzt ausgeführt wird, und Endgewichtsecken, für die die Eckenoperation zuletzt ausgeführt wird.

[0022] In einer weiteren Ausführungsform des Aufzeichnungsmediums weist jede Eckenliste zudem Daten über eigene Ecken auf, die nur von dem Modell beeinflusst werden, zu dem diese Eckenliste gehört.

[0023] In einer weiteren Ausführungsform des oberen Aufzeichnungsmediums weisen die Eckendaten Koordinatendaten und Normalendaten der Ecken auf.

[0024] In einer weiteren Ausführungsform des Aufzeichnungsmediums folgen die Eckendaten einem lokalen Koordinatensystem des Modells der Eckenliste, zu dem die Eckendaten gehören.

[0025] In einer weiteren Ausführungsform des obigen Aufzeichnungsmediums hat die Eckenkennung im gemeinsamen Eckenpuffer eine Eintragsnummer der Ecke eines jeden Modells sowie den Versatzwert, der der Anzahl der Ecken eines jeden Modells entspricht.

[0026] In einer weiteren Ausführungsform enthält jede der Polygonlisten einen Zeichnungsbefehl, der das Zeichnen des Polygons des Modells anweist, oder einen Nicht-Zeichnungsbefehl, der angibt, dass das Polygon des Modells nicht zu zeichnen ist, zu dem diese Polygonliste gehört.

[0027] Ein zweiter Aspekt der Erfindung stellt ein Bildverarbeitungsverfahren bereit, das ursprüngliche Modelldaten in integrierte Formgebungs-Modelldaten umwandelt, geeignet zum Modellieren der Gliederbewegung eines Objekts, wobei zahlreiche Modelle über eine hierarchische Struktur verknüpft sind, und mindestens ein erstes Modell zahlreiche Ecken aufweist, die Polygone bilden, und zumindest die Position einer ersten Ecke von den Positionen von zwei oder mehr der zahlreichen Modelle und von Gewichtungswerten aus diesen zwei oder mehr Modellen beeinflusst wird, und in den ursprünglichen Modelldaten zahlreiche Modelldaten über die hierarchische Struktur verknüpft sind, und jede der zahlreichen Daten umfassen: eine Liste ursprünglicher Ecken, die die Eckendaten dieser Modelldaten aufweist, eine Gewichtungsliste, die Eckendaten über Ecken aufweist, die von diesen Modelldaten beeinflusst werden, und eine Liste ursprünglicher Polygone, in der die Eckenkennung der Eckenliste zugeordnete Daten sind, und das Bildverarbeitungsverfahren die Schritte umfasst: das Erzeugen von Formatdaten eines gemeinsamen Eckenpuffers, damit Daten über die zahlreichen Ecken in den zahlreichen Modellen für jedes Modell in der hierarchischen Struktur gespeichert werden; das Erzeugen einer Eckenliste, die für jedes der Modelle erzeugt wird, das die Ecken beeinflusst, und in der Eckendaten über eine Eckenkennung im gemeinsamen Eckenpuffer festgelegt sind; und das Erzeugen einer Polygonliste, die für jedes der Modelle erzeugt wird, das die Polygone aufweist, bei denen die Eckenkennung zugeordnete Daten sind, wobei die Eckendaten in jeder Eckenliste umfassen: mindestens die Positionsdaten der Ecken, Gewichtungswerte aus dem Modell, zu dem die Eckenliste gehört, und Eckenkennungen im gemeinsamen Eckenpuffer, die diesen Ecken zugeordnet sind, und wobei der Schritt des Erzeugens einer Eckenliste für jedes Modell, das die Ecken beeinflusst, enthält: das Erzeugen von Eckendaten für die erste Ecke für jedes der zwei oder mehr der zahlreichen Modelle, das die erste Ecke beeinflusst; und das Verteilen der erzeugten Eckendaten auf die jeweiligen Eckenlisten der zwei oder mehr Modelle als verteilte Eckendaten.

[0028] Gemäß dem obigen Bildverarbeitungsverfahren kann man Konvertierungsdaten für ein integriertes Formgebungsmodell erzeugen, die eine Bildverarbeitung in Echtzeit erlauben.

[0029] In einer Ausführungsform des obigen Bildverarbeitungsverfahrens umfasst der Schritt des Erzeugens jeder der Eckenlisten, und zwar für jede Eckenliste, die verteilte Eckendaten enthält, das Unterteilen der Eckendaten in Daten für Anfangsgewichtsecken, für die eine Eckenoperation zuerst ausgeführt wird, wenn die hierarchische Struktur der Modelle durchlaufen wird, Daten für Mittelgewichtse-

cken, für die die Eckenoperation nicht zuerst oder zuletzt ausgeführt wird, und Daten für Endgewichtsecken, für die die Eckenoperation zuletzt ausgeführt wird, in der Eckenliste.

[0030] In einer weiteren Ausführungsform des obigen Bildverarbeitungsverfahrens hat die Eckenkennung eine Eintragsnummer der Ecke in jedem Modell, sowie einen Versatzwert, der der Anzahl der Ecken eines jeden Modells entspricht.

[0031] In einer weiteren Ausführungsform des obigen Bildverarbeitungsverfahrens weist das Bildverarbeitungsverfahren zudem einen Zeichnungsbefehl-Erzeugungsschritt auf, in dem: eine Dummy-Eckenoperation für die Eckendaten in der Eckenliste eines jeden Modells gemäß der Abarbeitungsfolge ausgeführt wird, die der hierarchischen Struktur der Modelle folgt; wenn die Dummy-Eckenoperation für die Eckenliste eines Modells endet, ein Nicht-Zeichnungsbefehl in der Polygonliste dieses Modells erzeugt wird, falls die Gewichtungsberechnung für die Ecken dieses Modells noch nicht beendet ist; und ein Zeichnungsbefehl in der Polygonliste dieses Modells erzeugt wird, falls die Gewichtungsberechnung für die Ecken des Nicht-Zeichnungsmodells beendet ist.

[0032] In einer weiteren Ausführungsform des genannten Bildverarbeitungsverfahrens weist das Bildverarbeitungsverfahren zudem einen Erzeugungsschritt für eine Liste einfach gewichteter Ecken auf, wobei die erste Ecke mit den verteilten Eckendaten, die von den zwei oder mehr der zahlreichen Modelle beeinflusst wird, in eine einfach gewichtete Ecke umgewandelt wird, die nur von einem Modell beeinflusst wird, das von den zwei oder mehr Modellen den größten Einfluss ausübt, und die Daten bezüglich der umgewandelten einfach gewichteten Ecke auf die Eckenliste dieses Modells verteilt werden.

[0033] Ein dritter Aspekt der Erfindung stellt ein Bildverarbeitungsverfahren zum Zeichnen eines integrierten Formgebungsmodells bereit, das zum Modellieren der Gliederbewegung eines Objekts geeignet ist, wobei zahlreiche Modelle über eine hierarchische Struktur verknüpft sind, und mindestens ein erstes Modell zahlreiche Ecken aufweist, die Polygone bilden, und zumindest die Position einer ersten Ecke von den Positionen von zwei oder mehr der zahlreichen Modelle und von Gewichtungswerten aus diesen zwei oder mehr Modellen beeinflusst wird, und die Daten des integrierten Formgebungsmodells umfassen:

Formatdaten eines gemeinsamen Eckenpuffers, der Daten über die zahlreichen Ecken in den zahlreichen Modellen speichert, und zwar für jedes Modell in der hierarchischen Struktur; eine Eckenliste, die für jedes der Modelle erzeugt

wird, das die Ecken beeinflusst, und in der Eckendaten über eine Eckenkennung im gemeinsamen Eckenpuffer festgelegt sind; und eine Polygonliste, die für jedes der Modelle erzeugt wird, das die Polygone aufweist, und die Polygondaten enthält, bei denen die Eckenkennung zugeordnete Daten sind,

wobei die Eckendaten in jeder Eckenliste umfassen: mindestens die Positionsdaten einer Ecke, einen Gewichtungswert aus dem Modell, zu dem die Eckenliste gehört, und die Eckenkennung im gemeinsamen Eckenpuffer, die der Ecke zugeordnet ist; und das Bildverarbeitungsverfahren die Schritte umfasst:

das Erzeugen des gemeinsamen Eckenpuffers entsprechend der zahlreichen Modelle in der Abarbeitungsfolge der hierarchischen Struktur gemäß den Formatdaten;

das Erzeugen von Modellmatrixdaten, in deren die Positionen der Modelle ausgehend von Spielfortschrittsdaten eingestellt werden;

das Erzeugen von gemeinsamen Eckendaten durch das Ausführen der Matrixberechnung zum Erzeugen von Eckendaten nach der Bewegung gemäß den Modellmatrixdaten, und einer Gewichtungsberechnung zum Integrieren von Gewichtungswerten aus den Modellen in die Eckendaten nach der Bewegung, und zwar für die Eckendaten der Eckenliste eines jeden Modells, und durch das Speichern oder Addieren dieser berechneten Eckendaten in Bereichen entsprechend den Eckenkennungen im gemeinsamen Eckenpuffer; und

das Rendern der Polygone gemäß den gemeinsamen Eckendaten; dadurch gekennzeichnet, dass Eckendaten der ersten Ecke, die von Positionen der zwei oder mehr der zahlreichen Modelle beeinflusst werden, auf die Eckenlisten der zwei oder mehr Modelle als verteilte Eckendaten aufgeteilt werden.

[0034] Gemäß dem obigen Bildverarbeitungsverfahren kann man die Figuren der integrierten Formgebungsmodelle in Echtzeit zeichnen.

[0035] In einer Ausführungsform des obigen Bildverarbeitungsverfahrens werden im Erzeugungsschritt der gemeinsamen Eckendaten die Positionsdaten für die obige Matrixberechnung mit den Modellmatrixdaten multipliziert, und die obigen multiplizierten Positionsdaten werden für die obige Gewichtungsberechnung mit dem Gewichtungswert multipliziert.

[0036] In einer weiteren Ausführungsform des obigen Bildverarbeitungsverfahrens wird jede der Eckenlisten unterteilt in: Anfangsgewichtsecken, für die eine Eckenoperation zuerst ausgeführt wird, wenn die Abarbeitung anhand der hierarchischen Struktur der Modelle erfolgt, Mittelgewichtsecken, für die die Eckenoperation nicht zuerst oder zuletzt ausgeführt wird, und Endgewichtsecken, für die die

Eckenoperation zuletzt ausgeführt wird, und im Erzeugungsschritt der gemeinsamen Eckendaten werden die berechneten Eckendaten für die Anfangsgewichtsecken in Bereichen gespeichert, die den Eckenkennungen im gemeinsamen Eckenpuffer entsprechen, und die berechneten Eckendaten für die Mittelgewichtsecken und die Endgewichtsecken werden zu den Eckendaten in Bereichen addiert, die den Eckenkennungen im gemeinsamen Eckenpuffer entsprechen.

[0037] In einer weiteren Ausführungsform des obigen Bildverarbeitungsverfahrens weist jede der obigen Polygonlisten einen Zeichnungsbefehl auf, der das Zeichnen der Polygone des Modells anweist, zu dem die Polygonliste gehört, oder einen Nicht-Zeichnungsbefehl, der angibt, dass die Polygone des Modells, zu dem die Polygonliste gehört, nicht zu zeichnen sind, und der Renderschritt beruht auf der Polygonliste des Modells, und zwar jedesmal, wenn der Erzeugungsschritt der obigen gemeinsamen Eckendaten eines jeden Modells endet, und das genannte Rendern wird gemäß dem Zeichnungsbefehl ausgeführt.

[0038] In einer weiteren Ausführungsform des obigen Bildverarbeitungsverfahrens umfasst das Verfahren zudem einen Freigabeschritt des Bereichs des gemeinsamen Eckenpuffers, der zu dem Modell gehört, für das das Rendern ausgeführt wird.

[0039] In einer weiteren Ausführungsform des obigen Bildverarbeitungsverfahrens weisen die angesprochenen Modelldaten auch eine Liste einfach gewichteter Ecken auf, in der die erste Ecke, die von den zwei oder mehr der zahlreichen Modelle beeinflusst wird, in eine einfach gewichtete Ecke umgewandelt wird, die nur von dem Modell beeinflusst wird, das von den zwei oder mehr Modellen den größten Einfluss ausübt, und die Daten bezüglich der einfach gewichteten Ecke auf die Eckenliste dieses Modells verteilt werden, und das genannte Bildverarbeitungsverfahren erstellt die gemeinsamen Eckendaten für ein vorbestimmtes Modell anhand der Liste einfach gewichteter Ecken.

[0040] Zum Erfüllen der beschriebenen Aufgaben stellt die Erfindung ein Aufzeichnungsmedium bereit, das ein Programm für einen Computer speichert, damit das angesprochenen Bildverarbeitungsverfahren ausgeführt wird. Mit diesem Aufzeichnungsmedium können die Figuren eines integrierten Formgebungsmodells in Spieldaten in Echtzeit gezeichnet werden.

KURZE BESCHREIBUNG DER ZEICHNUNGEN

[0041] Es zeigt:

[0042] Fig. 1 eine Zeichnung mit einem Beispiel für

die Bewegung eines integrierten Formgebungsmodells in CG-Spielfilmen;

[0043] [Fig. 2](#) eine Zeichnung einer herkömmlichen Bildverarbeitung, in der ein integriertes Formgebungsmodell in CG-Spielfilmen verwendet wird;

[0044] [Fig. 3](#) ein Flussdiagramm der Bildverarbeitungsprozedur in [Fig. 2](#);

[0045] [Fig. 4](#) eine Zeichnung mit einem Beispiel eines integrierten Formgebungsmodells, das in der Erfindung mit Hilfe des dreidimensionalen Modellierers erzeugt wird;

[0046] [Fig. 5](#) ein Beispiel der Datenkonfiguration des integrierten Formgebungsmodells in [Fig. 4](#);

[0047] [Fig. 6](#) eine Zeichnung, die den Einflussgrad (Gewichtungswert) vom Modell auf jede Ecke angibt, den ein Entwickler eingestellt hat;

[0048] [Fig. 7](#) eine Zeichnung, die ein Beispiel der Bewegung eines integrierten Formgebungsmodells darstellt (Hülle);

[0049] [Fig. 8](#) eine Zeichnung, die das Gewichtungsberechnungsverfahren in einem integrierten Formgebungsmodell darstellt;

[0050] [Fig. 9](#) ein Flussdiagramm, das die Prozedur der Modellbewegungsberechnung und der Gewichtungsberechnung von Ecken in einem integrierten Formgebungsmodell darstellt;

[0051] [Fig. 10](#) eine Zeichnung, die den Zusammenhang zwischen den Ursprungsdaten eines integrierten Formgebungsmodells und den Konvertierungsdaten für die Zeichnungsbibliothek in dieser Ausführungsform darstellt;

[0052] [Fig. 11](#) eine Zeichnung, die den Zusammenhang zwischen der Anordnung im Aufzeichnungsmedium für Spiele und der Spielemaschinen-Hardware darstellt;

[0053] [Fig. 12](#) ein Flussdiagramm, das die Abläufe zum Erzeugen von Konvertierungsdaten aus den integrierten Formgebungsmodelldaten durch einen dreidimensionalen Modellierer darstellt;

[0054] [Fig. 13](#) ein Flussdiagramm in einem Spiel;

[0055] [Fig. 14](#) ein Flussdiagramm, das eine Prozedur eines Konvertierers darstellt;

[0056] [Fig. 15](#) ein Flussdiagramm, das eine Prozedur eines Konvertierers darstellt;

[0057] [Fig. 16](#) ein Flussdiagramm, das eine Proze-

dur der Einzelheiten der Verteilung der tatsächlichen Ecken auf Modelle und die Eckenlistenerzeugung wiedergibt;

[0058] [Fig. 17](#) eine Zeichnung einer Prozedur zum Umsetzen der Ursprungsdaten in die Konvertierungsdaten;

[0059] [Fig. 18](#) eine Zeichnung der Eckenliste der Ursprungsdaten und der gemeinsamen Eckenliste, wenn die Erfindung auf ein bestimmtes Modellbeispiel angewendet wird;

[0060] [Fig. 19](#) eine Zeichnung einer Eckenliste für den Fall, dass die Erfindung auf ein bestimmtes Modellbeispiel angewendet wird;

[0061] [Fig. 20](#) eine Zeichnung mit einem Beispiel einer Polygonliste;

[0062] [Fig. 21](#) eine Zeichnung mit einem Beispiel einer Nicht-Zeichnungstabelle;

[0063] [Fig. 22](#) eine Zeichnung, die das Öffnen und die Freigabe eines gemeinsamen Eckenpuffers darstellt;

[0064] [Fig. 23](#) eine Zeichnung eines weiteren Beispiels einer Eckenliste;

[0065] [Fig. 24](#) ein Flussdiagramm einer Prozedur der Zeichnungsbibliothek; und

[0066] [Fig. 25](#) ein Flussdiagramm einer Prozedur des Konvertierers, mit der die einfach gewichtete Eckenliste aus den Daten eines mehrfach gewichteten integrierten Formgebungsmodells gewonnen wird.

BESCHREIBUNG DER BEVORZUGTEN AUSFÜHRUNGSFORMEN

[0067] Anhand der beiliegenden Zeichnungen werden nun Ausführungsformen der Erfindung beschrieben. Diese Ausführungsformen schränken jedoch den technischen Bereich der Erfindung nicht ein, der in den beigefügten Ansprüchen bestimmt ist.

[0068] Die Zeichnung in [Fig. 4](#) stellt ein Beispiel eines integrierten Formgebungsmodells dar, das gemäß der Erfindung mit Hilfe eines dreidimensionalen Modellierers erzeugt wird. Das integrierte Formgebungsmodell in [Fig. 4](#) ist ein Beispiel für die Implementierung der Bewegung des in [Fig. 1](#) dargestellten Arms. Das Modell M1 weist ein Polygon auf, das tatsächlich dargestellt wird. Dieses Polygon besitzt insgesamt 13 tatsächliche Ecken, nämlich V101–V113. Das Modell M1 ist beispielsweise eine Hülle, die den Oberarm 3 und das Unterarmteil 4 in [Fig. 1](#) einschließt. Die Position des Modells M1 besitzt z. B.

100 Prozent Einfluss auf die Ecken V101 und V107. Damit ist das Modell M1 ein Knochenmodell, das Gewichtungswerte für die tatsächlichen Ecken aufweist.

[0069] Die Modelle M2, M3 und M4 haben keine Polygone; daher besitzen sie keine tatsächlichen Ecken. Die Bewegung der Modelle M2, M3 und M4 beeinflusst jedoch einen Teil der Ecken des Modells M1 über vorbestimmte Gewichtungswerte. Anders ausgedrückt sind die Modelle M2, M3 und M4 Knochenmodelle, die Gewichtungswerte für tatsächliche Ecken haben. Zum Modell 5 gehört ein Polygon, das dem Handteil 2 in [Fig. 1](#) entspricht, und die sechs tatsächlichen Ecken V501–V506, die das Polygon bilden, werden nicht von anderen Modellen beeinflusst, sondern sie werden zu 100 Prozent von der Position des Modells M5 selbst beeinflusst. Damit ist das Modell M5 ein Knochenmodell, das seine eigenen Ecken V501–V506 beeinflusst.

[0070] Das integrierte Formgebungsmodell in [Fig. 4](#) wird auch als "Hülle" bezeichnet, die der Spielentwickler mit Hilfe eines dreidimensionalen Modellierers entwirft. Der Entwickler gestaltet den Entwurf so, dass jede Ecke von einem oder mehreren Modellen mit einem vorbestimmten Verhältnis beeinflusst wird. Durch die Bewegung der Position eines jeden Modells im Zuge der Ausführung des Spieleprogramms werden die Positionen der Ecken entsprechend der Einflussgrade (Gewichtungswerte) bewegt, die als Anfangswerte definiert sind und die gewünschte Bewegung der Figuren wird implementiert.

[0071] [Fig. 5](#) zeigt ein Datenkonfigurationsbeispiel des integrierten Formgebungsmodells in [Fig. 4](#). Die Modelle M2, M3 und M5 sind unter dem Modell M1 verbunden, und das Modell M4 ist ebenfalls unter dem Modell M1 verbunden. Diese Art von Baumstruktur der Daten ist ein Beispiel für die hierarchische Modellstruktur, und man kann die Modelle, die ein integriertes Formgebungsmodell bilden, innerhalb einer hierarchischen Struktur verbinden, die für das jeweilige Programm geeignet ist. Im Fall der in [Fig. 5](#) dargestellten Baumstruktur werden die Modelldaten generell in der Folge der Modelle M1, M2, M3, M5 und M4 abgearbeitet. Eine solche Abarbeitungsfolge kann jede beliebige Sequenz sein, die für das Programm geeignet ist.

[0072] Im Datenkonfigurationsbeispiel in [Fig. 5](#) besitzt jedes Modell eine Modellmatrix, die angibt, wie stark sich das Koordinatensystem des Modells bezüglich des Koordinatensystems des Modells in der höheren Hierarchiestufe bewegen sollte. Anders ausgedrückt enthält die Modellmatrix Daten zum Einstellen der Modellposition. Das Koordinatensystem wird später beschrieben; in diesem Beispiel hat jedoch jedes Modell das entsprechende lokale Koordinatensystem. Es existiert auch ein globales Koordinatensystem, das für alle Modelle gemeinsam gilt und von

den lokalen Koordinatensystemen getrennt ist. Zum Ermitteln einer Position des Modells M2 nach der Bewegung wird beispielsweise die Modellmatrix, die Drehung, Verschiebung und Skalierung enthält, als Positionszusammenhang des lokalen Koordinatensystems des Modells M2 bezüglich des lokalen Koordinatensystems des Modells M1 verwendet. Daher werden durch das Ausführen der Matrizenoperation, die die Modellmatrix mit den Positionskoordinaten und Normalendaten des lokalen Koordinatensystems des Modells M2 multipliziert, die Positionskoordinaten und Normalendaten des lokalen Koordinatensystems des Modells M2 in die Positionskoordinaten und Normalendaten des lokalen Koordinatensystems des Modells M1 konvertiert. Wird die Modellmatrix des Modells M1 mit den Positionskoordinaten und Normalendaten des Modells M1 multipliziert, so kann man die Positionskoordinaten und Normalendaten im globalen Koordinatensystem bestimmen.

[0073] Auf diese Weise kann sich eine Figur, die aus zahlreichen Modellen besteht, durch die Modellmatrix zwischen den Modellen bewegen. In der Regel wird eine solche Modellmatrix, die eine Bewegung einer Figur beschreibt, durch die Geometriefunktion im Spielprogramm erzeugt. Die Modellmatrix kann den Positionszusammenhang zwischen dem globalen Koordinatensystem und dem lokalen Koordinatensystem des Modells angeben.

[0074] Die Zeichnung in [Fig. 6](#) zeigt den Einflussgrad (Gewichtungswert) eines Modells auf jede Ecke, den ein Entwickler einstellt. Im Beispiel in [Fig. 6](#) werden die Ecken V101 und V107 des Modells M1 zu 100 Prozent von der Bewegung des Modells M1 beeinflusst. Daher bewegen sich die Positionen der Ecken V101 und V107 und behalten dabei den Positionszusammenhang mit der Bewegung des Knochenmodells des Modells M1. Die Ecken V102 und V103 des Modells M1 werden von den Modellen M2, M3 und M4 gemäß dem Verhältnis der Gewichtungswerte in [Fig. 6](#) beeinflusst. Damit werden diese Ecken mit der Größe der in [Fig. 6](#) dargestellten Gewichtungswerte beeinflusst, wenn sich die Modelle M2, M3 und M4 bewegen. Bewegt sich beispielsweise das Modell M4 um eine längere Strecke nach oben, so werden diese Ecken mit 40 Prozent der Bewegungsentfernung beeinflusst. Gehören die Ecken V102 und V103 zum Muskelteil 3A des Oberarmteils, so kann beispielsweise der Muskelteil 3A des Oberarmteils geeignet angehoben werden, indem das Modell M4 um die geeignete Entfernung bewegt wird.

[0075] Die Ecken V104, V105 und V108–V111 des Modells M1 werden von den Modellen M2 und M3 gemäß dem Verhältnis der Gewichtungswerte in [Fig. 6](#) beeinflusst. Die Ecken V106 und V112 werden nur vom Modell M3 beeinflusst. Bewegt sich das Modell M3, so bewegen sich also die Ecken V106 und V112,

und sie behalten dabei den gleichen Positionszusammenhang. Die Ecke V113 wird vom Modell M4 mit einem Gewichtungswert von 100 Prozent beeinflusst.

[0076] Das Modell M5 ist ein Teil, das mit dem Modell M1 verbunden ist, und entspricht dem Handteil **2** im Beispiel in [Fig. 1](#). Das Handteil **2** braucht seine Form abhängig von der Armposition nicht zu ändern und wird zu 100 Prozent vom Knochenteil des Modells M5 beeinflusst.

[0077] Den Gewichtungswert einer jeden in [Fig. 6](#) dargestellten Ecke setzt der Entwickler, der die Figur entwirft, als Anfangswert fest. Der Entwickler kann die Position einer jeden Ecke frei steuern, indem er derartige Gewichtungswerte einstellt und die Modelle während des Spielfortschritts durch die Modellmatrix bewegt. Sind die Positionen der Ecken festgelegt, so kann ein Polygon, das durch mehrere Ecken bestimmt ist, vom Zeichnungsabschnitt (Renderabschnitt) im Bildspeicher gerendert werden.

[0078] Die Zeichnung in [Fig. 7](#) stellt ein Beispiel der Bewegung eines integrierten Formgebungsmodells (Hülle) dar. Dieses Beispiel zeigt ein integriertes Formgebungsmodell für den Fall, dass sich das integrierte Formgebungsmodell in der in [Fig. 4](#) dargestellten Grundform bewegt, siehe [Fig. 1B](#). Zum Implementieren der in [Fig. 1B](#) dargestellten Bewegung bewegen sich die das integrierte Formgebungsmodell aufbauenden Modelle M1–M4 wie folgt. Die Modelle M1 und M2 drehen sich in vertikaler Richtung und das Modell M3 dreht sich in horizontaler Richtung. Gemeinsam damit bewegt sich das Modell M5 so, dass es mit der Spitzenposition des Modells M3 zusammenfällt. Das Modell M4 ist ein Modell, das hauptsächlich dazu dient, das Anschwellen **3A** des Muskels am Oberarm auszudrücken, und dreht sich vertikal nach oben, siehe [Fig. 7](#).

[0079] Dadurch bewegen sich die Ecken V102 und V103 so, dass der Muskel beeinflusst durch die Bewegung des Modells M4 hervortritt, und die Ecke V113 bewegt sich verriegelt mit der Bewegung des Modells M4. Die Ecken V101 bzw. V107 bewegen sich einfach fest verbunden mit der Bewegung des Modells M1, und die Ecken V104, V105 und V108–V111 bewegen sich jeweils in Positionen, die von der Bewegung der Modelle M2 und M3 beeinflusst werden. Dadurch verändert sich das von den Ecken V104, V109 und V110 gebildete Polygon in die Form eines natürlichen Ellbogengelenks, wenn ein Arm abgewinkelt wird. Die Ecke V112 bewegt sich verriegelt mit der Bewegung des Modells M3. Jede Ecke des Modells M5 bewegt sich einfach entsprechend der Bewegung des Modells M5.

[0080] Werden Polygone, die den Raum zwischen jeder Ecke füllen, bezüglich der Positionen der Ecken nach der Bewegung gerendert, so ist der Oberarm **3**

in vertikaler Richtung abgesenkt, der Unterarm **4** ist in eine stärker waagrechte Richtung gebogen, der Muskel **3A** des Oberarms tritt hervor, und das Handteil **2** ist an die Spitze des Arms angeschlossen, siehe [Fig. 1B](#). Die Modelle M2, M3 und M4, die keine Polygone besitzen, haben keine tatsächlichen Ecken, und diese Modelle werden in den Bildern nicht dargestellt.

[0081] Die Zeichnung in [Fig. 8](#) stellt das Gewichtungsberechnungsverfahren in einem integrierten Formgebungsmodell dar. [Fig. 8](#) zeigt den Fall, dass das integrierte Formgebungsmodell im Grundformstatus in [Fig. 4](#) nach der Bewegung den Status in [Fig. 7](#) annimmt, und sie zeigt insbesondere den Zusammenhang zwischen den Ecken V102 und V103 und den Modellen M2, M3 und M4, die die Ecken V102 und V103 beeinflussen. [Fig. 9](#) zeigt ein Flussdiagramm für die Prozedur der Modellbewegungsrechnung (Matrixberechnung) und der Gewichtungsberechnung der Ecken in einem integrierten Formgebungsmodell.

[0082] [Fig. 8A](#) zeigt die Modelle M2, M3 und M4 im Grundformstatus. Wie oben erwähnt besitzen die Modelle M2, M3 und M4 die lokalen Koordinatensysteme CDM2, CDM3 und CDM4. Das globale Koordinatensystem CDGB, das für alle Modelle gemeinsam gilt, ist ebenfalls eingestellt. Im integrierten Formgebungsmodell in [Fig. 4](#) hat ein Entwickler die Ecken V102 und V103 so eingestellt, dass sie (mit Gewichtungswerten) von den Modellen M2, M3 und M4 beeinflusst werden. Anders ausgedrückt werden die in [Fig. 6](#) dargestellten Gewichtungswerte als Anfangswerte vorab eingestellt. Zum Ausführen der Gewichtungsberechnung werden die Ecken V102 und V103 jeweils in drei Ecken aufgeteilt (verteilte Ecken), die zu den Modellen M2, M3 und M4 gehören, die die Ecken jeweils beeinflussen, und sie werden auf die jeweils zugehörigen Modelle verteilt (S1 in [Fig. 9](#)). Beispielsweise wird die Ecke V102 in die verteilten Ecken V102 (M2), V102 (M3) und V102 (M4) unterteilt, die auf die Modelle M2, M3 und M4 verteilt werden, die jeweils die verteilten Ecken beeinflussen. Die Ecke V103 wird ebenfalls in drei verteilte Ecken unterteilt, die den Modellen zugeteilt werden, die jeweils die verteilten Ecken beeinflussen.

[0083] Dadurch gehören die Ecken V102 (M2) und V103 (M2) zum Modell M2. In gleicher Weise gehören die Ecken V102 (M3) und V103 (M3) zum Modell M3, und V102 (M4) und V103 (M4) gehören zum Modell M4. Die Positionskoordinaten der Ecken, die zu jedem Modell im Grundformstatus gehören, werden in das lokale Koordinatensystem eines jeden Modells umgewandelt (S2 in [Fig. 9](#)). Beispielsweise werden die absoluten Koordinaten der Ecke V102 (M2) und der Ecke V103 (M2) anhand der Modellmatrix des Modells M2 in das lokale Koordinatensystem CDM2 des Modells M2 umgewandelt. Für die anderen Ecken werden ebenfalls die absoluten Koordinaten in

das lokale Koordinatensystem des Modells umgewandelt, zu dem die Ecken gehören. Dadurch bewegt sich jede Ecke zusammen mit dem Modell, zu dem die Ecke gehört.

[0084] [Fig. 8B](#) zeigt einen Status, in dem sich jedes der Modelle M2, M3 und M4 wie in [Fig. 7](#) dargestellt bewegt. Die Bewegung eines jeden Modells wird durch das Einstellen der jeweiligen Modellmatrix gesteuert (S3 in [Fig. 9](#)). Beispielsweise bewegt sich das lokale Koordinatensystem CDM2 des Modells M2 in eine Position, die der Drehung, Verschiebung und Skalierung der Modellmatrix des Modells M2 bezüglich des lokalen Koordinatensystems des Modells M1 entspricht, das nicht dargestellt ist. Die zum Modell M2 gehörenden Ecken V102 (M2) und V103 (M2) verschieben sich gemäß der Bewegung des Modells M2 und behalten dabei die Positionen bezüglich des Modells M2 im Grundformstatus. Anders ausgedrückt bewegen sich die Positionen der Ecken V102 (M2) und V103 (M2) im lokalen Koordinatensystem CDM2 des Modells M2 nicht. Damit kann man die Positionen der Ecken V102 (M2) und V103 (M2), die sich zusammen mit der Bewegung des Modells M2 bewegen, auch dadurch bestimmen, dass man die Matrixberechnung mit Hilfe der Modellmatrix des Modells M2 ausführt (S4 in [Fig. 9](#)).

[0085] Die Ecken V102 (M3) und V103 (M3), die zum Modell M3 gehören, und die Ecken V102 (M4) und V103 (M4) die zum Modell M4 gehören, bewegen sich ebenfalls gemeinsam mit der Bewegung der jeweiligen Modelle, und man kann die Position nach der Bewegung dadurch bestimmen, dass man die Matrixberechnung mit Hilfe der Modellmatrix der jeweiligen Modelle ausführt.

[0086] Sind für die auf jedes Modell verteilten Ecken die Positionen im globalen Koordinatensystem ermittelt, so wird beispielsweise der Gewichtungsvorgang abhängig von den Gewichtungswerten (Einflussgrad) ausgeführt, die jedes Modell für die Ecke aufweist (S5 in [Fig. 9](#)). Gemäß den Gewichtungswerten in [Fig. 6](#) haben beispielsweise die Gewichtungswerte aus den Modellen M2, M3 und M4 die Werte 30 Prozent, 30 Prozent und 40 Prozent, und zwar jeweils für beide Ecken. Mit diesen Vorgaben sei beispielsweise die Ecke V102 betrachtet. Die neue Positionscoordinate der Ecke V102 gemäß dem Einflussgrad (Gewichtungswert) eines jeden Modells wird dadurch bestimmt, dass man den Eckencoordinatenwert der Ecke V102 (M2) mit 0,3 multipliziert, den Eckencoordinatenwert der Ecke V102 (M3) mit 0,3 multipliziert, den Eckencoordinatenwert der Ecke V102 (M4) mit 0,4 multipliziert, und diese Werte integriert (addiert). Anders ausgedrückt ist die durch die Doppelkreise in [Fig. 8\(B\)](#) bezeichnete Ecke V102 die Position der tatsächlichen Ecke, die aus den drei verteilten Ecken V102 (M2), V102 (M3) und V102 (M4) mit Hilfe der Gewichtungswerte 30 Prozent, 30 Prozent und 40

Prozent ermittelt wird. Sind die X-Koordinatenwerte der drei verteilten Ecken V102 (M2), V102 (M3) und V102 (M4) im globalen Koordinatensystem X2, X3 und X4, so erhält man den X-Koordinatenwert X102 der tatsächlichen Ecke V102, die durch den Doppelkreis dargestellt ist, durch:

$$X102 = 0,3 \cdot X2 + 0,3 \cdot X3 + 0,4 \cdot X4$$

(· bezeichnet eine Multiplikation); auf diese Weise erfolgt die Gewichtungsberechnung.

[0087] Unter Verwendung der Koordinatenwerte und der Normalendaten der Ecken, die mit der obigen Vorgehensweise gewonnen werden und nach der Gewichtungsberechnung mit Doppelkreisen dargestellt sind, erfolgt das Rendern der Polygone, die aus zahlreichen Ecken bestehen. Beispielsweise werden die Farbdaten von Pixeln in einem Polygon, das mit zahlreichen Ecken verbunden ist, dadurch ermittelt, dass man das innere Verhältnis der Pixelkoordinaten bezüglich der Eckencoordinaten mit einem Raster-Scan-Verfahren ermittelt und eine Interpolation mit Hilfe des inneren Verhältnisses abhängig von den Eckendaten und Texturdaten vornimmt. Zusätzlich zu den obigen Eckendaten werden auch Normalendaten bezüglich der Normalenvektoren der Ecke verwendet. Die Normalendaten an der mit Doppelkreisen bezeichneten Ecke werden durch einen Gewichtungsvorgang bestimmt, und zwar den gleichen Gewichtungsvorgang wie oben, und die Berechnung von verteiltem Licht bezüglich der Lichtquelle erfolgt mit Hilfe dieser Normalendaten. Diese Renderverfahren sind jedoch nicht Gegenstand der Erfindung, in der ein allgemein übliches Renderverfahren verwendet wird.

[0088] Oben ist das Verfahren beschrieben worden, das die Bewegung einer Figur mit dem integrierten Formgebungsmodell (Hülle) mit Hilfe eines dreidimensionalen Modellierers implementiert. Wird ein integriertes Formgebungsmodell verwendet, so muss jedes Modell wie bereits erklärt zusätzlich zu den tatsächlichen Eckendaten des jeweiligen Modells Daten über die Einflussecken aufweisen. Zudem ist der Schritt zum Bestimmen der Position eines jeden Modells durch eine Operation mit Hilfe der Modellmatrix, der Schritt zum Bestimmen der Positionen der Einflussecken mit Hilfe der Modellmatrix, der Schritt zum Bestimmen der neuen Position und Normalendaten unter Beachtung des Einflussgrades dieser Ecken durch die Gewichtungsberechnung und der Schritt, in dem das Rendern der Polygone eines jeden Modells mit Hilfe der oben ermittelten Eckendaten erfolgt, nötig. Die Verarbeitung des integrierten Formgebungsmodells, die drei Schritt umfasst, muss in Echtzeit und auf einer Hardware erfolgen, beispielsweise einer Spieldatenmaschine, die nur einfache Betriebsfunktionen aufweist und einen kleinen Speicherumfang hat, und nicht auf einem Supercomputer.

[0089] In der Erfindung werden Konvertierungsdaten verwendet, die man dadurch erhält, dass ein Teil der obigen Datenverarbeitung vorab an den ursprünglichen Modelldaten des integrierten Formgebungsmodells ausgeführt wird. Diese Konvertierungsdaten besitzen ein Format, das für den Betrieb und für die Verarbeitung durch die Spieldatenmaschine in Echtzeit geeignet ist. Diese Konvertierungsdaten werden in einem Aufzeichnungsmedium für eine Spieldatenmaschine als Figurendaten aufgezeichnet, z. B. einer CD-ROM oder Spieldatenkassette.

[0090] Die Zeichnung in [Fig. 10](#) gibt den Zusammenhang zwischen den Ursprungsdaten eines integrierten Formgebungsmodells und den Konvertierungsdaten für die Zeichnungsbibliothek in dieser Ausführungsform wieder. Gemäß dieser Ausführungsform werden die Ursprungsdaten **100** des integrierten Formgebungsmodells, die ein dreidimensionaler Modellierer erzeugt, wie er für einen CG-Spielfilm verwendet wird, in Konvertierungsdaten **110** für die Zeichnungsbibliothek umgewandelt, und zwar von einem neuen Konvertierungsprogramm, das die Erfindung bereitstellt. Die Konvertierungsdaten **110** enthalten keine Daten, die sich während des Spielfortschritts dynamisch ändern. Sie besitzen ein Datenformat, das für das Ausführen der dynamischen Bewegung von Figuren während des Spielfortschritts durch die Zeichnungsbibliothek (Zeichnungsverarbeitungsprogramm) optimiert ist, und zwar in einem Aufzeichnungsmedium, in dem ein Spieldatenprogramm gespeichert ist, das die Spieldatenmaschine ausführt. Die Konvertierungsdaten **110** werden dadurch erzeugt, dass die Ursprungsdaten mit dem Konvertierungsprogramm umgewandelt und in einem Aufzeichnungsmedium für Spiele gespeichert werden, beispielsweise einer CD-ROM.

[0091] Die Zeichnung in [Fig. 11](#) stellt den Zusammenhang zwischen der Anordnung im Aufzeichnungsmedium für Spiele (z. B. CD-ROM) und der Hardware der Spieldatenmaschine dar. Das Aufzeichnungsmedium **10**, siehe [Fig. 11](#), dient dem Speichern eines Spieldatenprogramms **12**, das die Handlung des Spiels und die Bewegung der Objekte einschließlich der Figuren steuert und sie mit dem Spielfortschritt synchronisiert. Es enthält auch Modellkonvertierungsdaten **110**, die der Konvertierer durch Umwandeln der Ursprungsdaten erzeugt hat, und eine Zeichnungsbibliothek **14** (eine Art von Programm), die die Bildverarbeitung ausführt. Ein derartiges Aufzeichnungsmedium **10** für Spiele ist in die Hardware einer Spieldatenmaschine **20** eingebaut. Beim Ausführen des Spieldatenprogramms wird das Aufzeichnungsmedium bei Bedarf gelesen, und die Zeichnungsbibliothek **12** wird mit Hilfe der Modellkonvertierungsdaten ausgeführt. Dadurch werden Bilder innerhalb kurzer Zeitintervalle erzeugt, beispielsweise in jedem Frame.

[0092] [Fig. 11](#) zeigt eine Anordnung einer Spieldaten-

maschine-Hardware **20**. Die CPU **22**, die die arithmetische Verarbeitung ausführt, und ein Systemspeicher **24** (RAM), in den zeitweilig das Programm des Aufzeichnungsmediums **10** für Spiele und die Modellkonvertierungsdaten eingelesen werden, und der als Pufferspeicher für die arithmetische Verarbeitung dient, sind in der Spieldatenmaschine-Hardware **20** über Busse verbunden. Über einen Busverwalter **38** sind auch verbunden: ein Renderprozessor **26**, der das Rendern gemäß den Zeichnungsbefehlen ausführt, die die Zeichnungsbibliothek erzeugt, ein Framespeicher **28**, in dem die Bildsignale aufgezeichnet werden, die der Prozessor **26** zeichnet, ein Audioprozessor **30**, der Audiodaten erzeugt, und ein Audiospeicher **32**, der die erzeugten Audiodaten aufzeichnet. Die im Framespeicher **28** abgelegten Bilddaten und die im Audiospeicher **32** aufgezeichneten Audiodaten werden in der Anzeigevorrichtung **34**, beispielsweise einem Fernsehmonitor, dargestellt oder von ihr ausgegeben. Der Busverwalter **38** in der Spieldatenmaschine **20** hat auch die Funktion einer Schnittstelle, über die das Steuergerät **42** für Spieldateneingaben als Eingabevorrichtung angeschlossen ist. Daran angeschlossen ist auch eine externe Kommunikationsleitung über ein Modem **40**. Der Busverwalter **38** ist auch mit einem Boot-ROM **36** verbunden; dadurch erfolgt der Hochlaufvorgang beim Einschalten durch das Ausführen der Daten im Boot-ROM. Die Spieldatenmaschine **20** stellt über den Busverwalter **38** auch die Verbindung zum Spieldaten-Aufzeichnungsmedium **10** her, beispielsweise eine CD-ROM oder eine Spieldatenkassette, so dass das Aufzeichnungsmedium **10** als äußere Aufzeichnungsvorrichtung genutzt wird.

[0093] Die im Aufzeichnungsmedium **10** für Spiele aufgezeichneten Modellkonvertierungsdaten **110** wurden in ein Datenformat umgewandelt, das die Bildverarbeitungslast der Zeichnungsbibliothek **14** verringert. Aufgrund dieses Datenformats, das später beschrieben wird, weisen die Modellkonvertierungsdaten **110** eine optimale Datenstruktur für die Zeichnungsbibliothek **14** auf. Daher kann die Bildverarbeitung für die integrierten Formgebungsmodelle der Figuren, die sich während des Spielfortschritts bewegen, in Echtzeit ausgeführt werden.

[0094] [Fig. 12](#) zeigt ein Flussdiagramm der Prozeduren zum Erzeugen der Konvertierungsdaten aus dem integrierten Formgebungsmodell durch einen dreidimensionalen Modellierer. Das Spiel wird normalerweise durch einen Entwickler entworfen, der die Spielhandlung erzeugt, einen Designer, der die Spieldatenobjekte für das Spiel entwirft, beispielsweise die Figuren, und einen Programmierer, der das Spieldatenprogramm schreibt. Der Designer verwendet den dreidimensionalen Modellierer zum Bereitstellen von Figuren, die sich so natürlich wie möglich bewegen, und erzeugt Ursprungsdaten des integrierten Formgebungsmodells, die als Hülle bezeichnet werden (S10). In diesem integrierten Formgebungsmodell

weist jedes Modell wie beschrieben eine Modellmatrix, Eckendaten und eine Gewichtungsliste auf, wie dies für den Stand der Technik in [Fig. 2](#) dargestellt ist. Die Modellmatrix umfasst Matrixdaten über die Drehung, Verschiebung und Skalierung bezüglich eines Modells auf einer höheren Hierarchieebene in der hierarchischen Struktur (Baumstruktur) der Modelle. Die Eckendaten sind Daten, die den tatsächlichen Ecken entsprechen, und enthalten mindestens Positionskoordinatendaten und Normalendaten der Normalenvektoren der Ecken. Die Eckendaten enthalten auch einen Wert α , der beispielsweise die Undurchsichtigkeit und Texturkoordinaten bezeichnet, die zu den Texturdaten gehören. Die Gewichtungsliste weist zumindest die Kennungen der tatsächlichen Ecken auf, die das Modell beeinflusst (Modellkennung und Eckenkennungen, die zu dem Modell gehören), und Gewichtungswerte. [Fig. 6](#) zeigt die Daten, wenn die Liste aufgebaut ist.

[0095] Nun erzeugt das Konvertierungsprogramm die Konvertierungsdaten entsprechend dem Datenformat für die Zeichnungsbibliothek aus den Ursprungsdaten des integrierten Formgebungsmodells (S12). In den Konvertierungsdaten, deren Einzelheiten im Folgenden beschrieben werden, sind die tatsächlichen Ecken in vier Gruppen eingeteilt, nämlich eigene Ecken, die keine Gewichtung haben, Anfangsgewichtsecken, Mittelgewichtsecken und Endgewichtsecken, für die Gewichtungsberechnungen erforderlich sind. Die Anfangsgewichtsecke ist eine Ecke, für die der Gewichtungswert in der Gewichtungsberechnung zuerst multipliziert wird. Die Mittelgewichtsecke ist eine Ecke, für die der Gewichtungswert in der Gewichtungsberechnung danach multipliziert wird. Die Endgewichtsecke ist eine Ecke, für die der Gewichtungswert in der Gewichtungsberechnung zuletzt multipliziert wird. Die Abfolge der Gewichtungsberechnung wird in der Regel anhand der Abarbeitungsfolge der hierarchischen Struktur (Baumstruktur) der Modelle bestimmt. Die in jede Gruppe eingeteilten Ecken werden für jedes Modell, zu dem die entsprechenden Ecken gehören, sortiert.

[0096] Nun wird das Format des gemeinsamen Eckenpuffers erzeugt, der die eigenen Ecken und die Anfangsgewichtsecken enthält, die tatsächliche Ecken ohne Überlappung bezeichnen. Den im gemeinsamen Eckenpuffer angeordneten tatsächlichen Ecken werden konsistente Eckenkennungen unabhängig vom Attribut des Modells zugeordnet. Die Liste der tatsächlichen Ecken, die aus den eigenen Ecken und den Anfangsgewichtsecken im gemeinsamen Eckenpuffer besteht, ist eine Menge der tatsächlichen Ecken für jedes Modell, zu dem die tatsächlichen Ecken gehören. Die gemeinsame Eckenliste für jedes Modell wird für jedes Modell sortiert, das diese tatsächlichen Ecken beeinflusst, und sie wird in der Folge der Modelle sortiert. Die konsistenten Eckenkennungen werden der tatsächlichen Eckenliste für

jedes Modell zugewiesen, das in den gemeinsamen Eckenpuffer eingeordnet ist, damit die Referenzierungsprozedur in der Gewichtungsberechnung in einem späteren Schritt vereinfacht wird. Der auf diese Weise erzeugte gemeinsame Eckenpuffer weist eine einfache Datenstruktur auf, die äquivalent zu einer herkömmlichen tatsächlichen Eckenliste eines Modells ist, bei dem keine Gewichtungsberechnung erfolgt.

[0097] Zusätzlich zum Format des gemeinsamen Eckenpuffers werden die in vier Gruppen unterteilten Ecken auf die Modelle verteilt, die diese Ecken beeinflussen. Eine Eckenliste wird für jedes Modell erzeugt, die die verteilten Ecken enthält. In dieser Eckenliste werden die Ecken in die genannten vier Gruppen einsortiert, nämlich eigene Ecke, Anfangsgewichtung, Mittelgewichtung und Endgewichtung. In dieser Weise werden die Ecken abgespeichert. Die Eckendaten in der Eckenliste enthalten die Eckenkoordinatendaten abhängig von den lokalen Koordinaten im Modell, den Normalendaten und den Gewichtungsdaten, die sich bei Bedarf auf die Eckenkennung im gemeinsamen Eckenpuffer der tatsächlichen Einflussecken beziehen. Anders ausgedrückt sind die genannten Gewichtungsdaten in den Eckendaten enthalten, die zu den Anfangsgewichtsecken, den Mittelgewichtsecken und den Endgewichtsecken gehören, und sie sind nicht in den Eckendaten enthalten, die zu den eigenen Ecken gehören.

[0098] In der Eckenliste für jedes Modell sind die eigenen Ecken, die nur vom Modell beeinflusst werden (eine Gewichtung), und die Anfangsgewichtsecken, die Mittelgewichtsecken und die Endgewichtsecken, die vom Modell und von anderen Modellen beeinflusst werden (mehrere Gewichtungen) in eindeutiger Weise angeordnet. Die Zeichnungsbibliothek führt die Bildverarbeitung mit Hilfe der Eckenliste und des gemeinsamen Eckenpuffers in Echtzeit aus.

[0099] Die Modellkonvertierungsdaten enthalten auch die Polygonliste. Die Polygonliste enthält mehrere Eckendaten, die das Polygon bilden. Die den tatsächlichen Ecken im gemeinsamen Eckenpuffer zugewiesenen Eckenkennungen werden in den Eckendaten der Polygonliste als zugeordnete Daten (Attributdaten) verwendet. Bevorzugt enthält die Polygonliste auch einen Zeichnungsbefehl, der angibt, ob ein Rendern (Zeichnen) des Modells ausgeführt wird. Dieser Zeichnungsbefehl wird vom Konvertierer erzeugt, und die Zeichnungsbibliothek greift darauf zu.

[0100] Wie beschrieben haben die vom Konvertierer erzeugten Konvertierungsdaten **110** das Format des gemeinsamen Eckenpuffers und bezeichnen die Liste der tatsächlichen Ecken, eine Eckenliste der in jedem Modell verteilten Ecken und die Polygonliste eines jeden Modells. Die Konvertierungsprozedur durch den Konvertierer, siehe [Fig. 9](#), umfasst das

Verteilen der tatsächlichen Ecken auf das Modell (S1) und die Umwandlungen in das lokale Koordinatensystem im Modell (S2), anschließend die Bewegungsberechnung abhängig von der Modellmatrix (Matrizenoperation), die die Zeichnungsbibliothek ausführt (S3, S4), und das Erzeugen optimaler Konvertierungsdaten für die Gewichtungsoperation mit Hilfe der Gewichtungswerte (S5) wird vorgenommen. Die Konvertierungsdaten werden im Weiteren nochmals anhand der Beispiele in [Fig. 4](#) und [Fig. 7](#) beschrieben.

[0101] Zusammen mit der Erzeugung der Konvertierungsdaten **110** erzeugt ein Programmierer (S14) wie erwähnt das Spielprogramm. Das Spielprogramm **12**, siehe [Fig. 11](#), die Modellkonvertierungsdaten **110** und die Zeichnungsbibliothek **14** werden im Aufzeichnungsmedium **10** für Spiele aufgezeichnet (S16).

[0102] [Fig. 13](#) zeigt ein Flussdiagramm eines Spiels. Zum Spielen eines Spiels wird ein Spiele-Aufzeichnungsmedium **10** in die Spieldatenmaschine eingesetzt (S20). Die Spieldatenmaschine führt das Spiel anhand des Spielprogramms im Aufzeichnungsmedium aus und erzeugt eine Modellmatrix zum Bewegen der Modelle, die eine Figur bilden (S22). Die Bewegungs-Modellmatrix ist wie erwähnt eine Matrix, die Parameter wie Drehung, Verschiebung und Skalierung bezüglich des Modells auf der höheren Hierarchiestufe enthält. Neue Koordinaten nach der Bewegung werden durch das Multiplizieren des Modells oder der Eckenkoordinaten des Polygons mit der Modellmatrix ermittelt.

[0103] Nun führt die Spieldatenmaschine die Zeichnungsbibliothek aus (S24). Die Modellkonvertierungsdaten **110** haben das Format des gemeinsamen Eckenpuffers, in dem konsistente Kennungen für die tatsächlichen Ecken aller Modelle zugewiesen werden und eine Eckenliste der für jedes Modell verteilten Ecken sowie eine Polygonliste für jedes Modell. Die Eckenliste für jedes Modell besteht aus Ecken, die nicht von anderen Modellen beeinflusst werden (eigene Ecken), und aus Ecken die von anderen Modellen beeinflusst werden (Anfangsgewichtsecken, Mittelgewichtsecken und Endgewichtsecken), siehe unten, und die Ecken, die von anderen Modellen beeinflusst werden, werden für jedes andere einflussnehmende Modell sortiert. In der Eckenliste für jedes Modell werden die Eckenkennungen anderer einflussnehmender Ecken gemäß den konsistenten Eckenkennungen im gemeinsamen Eckenpuffer verwendet. Die Polygonliste für jedes Modell enthält Eckendaten, für die die Eckenkennungen der Polygone, die das Modell bilden, zugeordnete Daten sind, und sie enthält einen Zeichnungsbefehl für das Modell, für das die Gewichtungsberechnung beendet ist, und einen Nicht-Zeichnungsbefehl für das Modell, für das die Gewichtungsberechnung noch nicht beendet

ist. Der gemeinsame Eckenpuffer wird im Systemspeicher (RAM) der Spieldatenmaschine aufgebaut, und zwar in dem Stadium, in dem die Zeichnungsbibliothek die Bildverarbeitung vornimmt.

[0104] In der Prozedur für die Zeichnungsbibliothek werden neue Koordinatendaten der Ecken nach der Bewegung dadurch bestimmt, dass die Koordinatendaten der Ecken eines jeden Modells mit der Bewegungsmatrix multipliziert werden, die das Spielprogramm erzeugt. Dieser Schritt entspricht dem Schritt S4 in [Fig. 9](#). Daraufhin wird ausgehend von den Eckendaten die Gewichtungsberechnung gemäß der Verarbeitungsfolge der hierarchischen Struktur der Modelle und in der Folge der Eckenliste im Modell ausgeführt. Das Berechnungsergebnis wird in den Bereich im gemeinsamen Eckenpuffer geschrieben oder integriert (addiert), der zu den einflussnehmenden Ecken gehört. Dieser Schritt entspricht dem Schritt S5 in [Fig. 9](#). Nun führt die Zeichnungsbibliothek das Zeichnen (Rendern) der Polygone im Modell aus, wobei gemäß dem Zeichnungsbefehl in der Polygonliste in jedem Modell auf die Eckenkoordinaten und Normalendaten zurückgegriffen wird, die im gemeinsamen Eckenpuffer gespeichert sind, und die Zeichnungsbibliothek zeichnet die Bilddaten einschließlich der Farbdaten der Pixel in den Polygonen im Framepuffer **28** auf.

[0105] Die Bilder werden auf der Anzeigevorrichtung **34** entsprechend den Bilddaten dargestellt, die im Framepuffer **28** gezeichnet sind (S26). Durch das Darstellen dieser gezeichneten Bilder für jeden Frame kann man Bilder einschließlich der Figuren, die sich abhängig von den eingegebenen Steuersignalen der Bedienperson bewegen, anzeigen. Da ein Modell verwendet wird, das auf dem integrierten Formgebungsmodell beruht, das ein dreidimensionaler Modellierer erzeugt, bewegen sich die Gelenke der Figuren natürlich, und man kann Bewegungen wie das Hervortreten eines Muskels bei einer Figur darstellen. Die angesprochenen Schritte S20 bis S26 werden für jede Frameperiode wiederholt.

[0106] [Fig. 14](#) und [Fig. 15](#) zeigen Flussdiagramme, die Prozeduren des Konvertierers darstellen. Anders als beim Zeichnungsbefehl, der im Aufzeichnungsmedium für Spiele gespeichert ist, braucht der Konvertierer der Ausführungsform die Daten nicht in Echtzeit zu verarbeiten. Die Schritte S30, S32 und S34 dieses Flussdiagramms sind die Stufen, in denen das Format des gemeinsamen Eckenpuffers aus den Ursprungsdaten des Modells erzeugt wird, und in denen die Eckenliste für jedes Modell erstellt wird. Die Schritte S36 bis S60 dieses Flussdiagramms sind Stufen eines Simulationsvorgangs der Bildverarbeitung, die die Zeichnungsbibliothek ausführt. Das Ausführen dieses Simulationsvorgangs ermöglicht es, die zeitlichen Abläufe beim Ausführen des Zeichnens (Renderns) eines jeden Modells zu erkennen,

wenn die hierarchische Struktur der Modelle abgearbeitet wird, und einen Nicht-Zeichnungsbefehl und einen Zeichnungsbefehl in die Polygonliste des Modells einzufügen. Das Format des gemeinsamen Eckenpuffers und der Eckenliste, die auf diese Weise erzeugt werden, ist ein Datenformat, mit dem die Zeichnungsbibliothek die Bildverarbeitung ausführen kann. Der Nicht-Zeichnungsbefehl oder die Polygonliste mit einem Zeichnungsbefehl, die der Simulationsvorgang erzeugt, senken die Belastung der Zeichnungsbibliothek zum Verwalten der zeitlichen Zeichnungsabläufe eines jeden Modells.

[0107] Das Konvertierungsprogramm, das die Verarbeitungsprozedur in [Fig. 14](#) aufweist, wird von einem Allzweckcomputer ausgeführt, der schnellere Verarbeitungsfunktionen und eine höhere Speicherkapazität hat als eine Spielemaschine. Das Konvertierungsprogramm ist in einem externen Aufzeichnungsmedium dieses Allzweckcomputers aufgezeichnet, beispielsweise einer Festplatte, der die Verarbeitungsprozedur für die Ursprungsdaten ausführt und die Konvertierungsdaten erzeugt.

[0108] Zum Ausführen der Bewegungsberechnung (Matrixberechnung) und der Gewichtungsberechnung für das Zeichnen des integrierten Formgebungsmodells, siehe die Erklärung in [Fig. 9](#), ist es erforderlich, die tatsächlichen Ecken auf die einflussnehmenden Modelle zu verteilen (S1) und die Positionskoordinatendaten und Normalendaten der Ecken in das lokale Koordinatensystem des Verteilungszielmodells umzuwandeln (S2). Zusätzlich zu den genannten beiden Prozeduren S1 und S2 erzeugt der Konvertierer der Erfahrung die tatsächlichen Ecken für die Anzahl der Verteilungszielmodelle und unterteilt diese tatsächlichen Ecken in vier Gruppen, nämlich eigene Ecken, die von der Position eines einzigen Modells abhängen (Ecken, die keine Gewichtung aufweisen), Anfangsgewichtsecken, für die die Gewichtungsberechnung zuerst ausgeführt werden muss, wenn die Modellhierarchie abgearbeitet wird, Mittelgewichtsecken, für die die Gewichtungsberechnung an zweiter Stelle oder später ausgeführt werden muss, und Endgewichtsecken, für die die Gewichtungsberechnung zuletzt ausgeführt werden muss (die Anfangsgewichtsecken, die Mittelgewichtsecken und die Endgewichtsecken sind Ecken, die von mehreren Modellen beeinflusst werden (Ecken, die Gewichtungen aufweisen)) (S30).

[0109] Da die eigenen Ecken und Anfangsecken der vier Eckengruppen den tatsächlichen Ecken entsprechen, ordnet der Konvertierer die Ecken dieser Eckengruppen für jedes Modell, zu dem diese Ecken gehören, neu, und erzeugt das Format des gemeinsamen Eckenpuffers (S32). Diesen Ecken, die im gemeinsamen Eckenpuffer angeordnet sind, werden dadurch konsistente neue Eckenkennungen zugewiesen, dass man sie der Reihe nach durchnumme-

riert. Für eine neue Eckenkennung kann man eine Zahl in unterschiedlichen Formaten verwenden. Beispielsweise werden eine Eintragsnummer der Ecke in jedem Modell und ein Versatzwert verwendet, der der Anzahl der Ecken im Modell entspricht, damit man die Formatdaten des gemeinsamen Eckenpuffers erzeugen kann, und man kann die Zahl auch für die Eckenkennung verwenden.

[0110] Der Konvertierer ordnet auch die vier Eckengruppen um, die auf die Modelle verteilt sind, und zwar für jedes Modell, das diese Ecken beeinflusst, und erzeugt die Eckenliste für jedes Modell (S34).

[0111] Die angesprochenen Speicherverarbeitungen S30, S32 und S34 werden im Weiteren erklärt, wobei die Modellbeispiele in [Fig. 4](#) und [Fig. 6](#) verwendet werden. [Fig. 16](#) zeigt ein Flussdiagramm der ausführlichen Prozedur zum Verteilen der tatsächlichen Ecken auf Modelle und zum Erzeugen der Eckenliste. Die Zeichnung in [Fig. 17](#) zeigt die Prozedur zum Umwandeln von Ursprungsdaten in Konvertierungsdaten. Die Zeichnung in [Fig. 18](#) zeigt eine Eckenliste der Ursprungsdaten und eine gemeinsame Eckenliste, wenn die Ausführungsform auf ein bestimmtes Modellbeispiel angewendet wird. Die Zeichnung in [Fig. 19](#) zeigt eine Eckenliste für den Fall, dass die Ausführungsform auf ein bestimmtes Modellbeispiel angewendet wird. Die Prozedur in [Fig. 16](#) und [Fig. 17](#) wird nun anhand von [Fig. 18](#) und [Fig. 19](#) beschrieben.

[0112] Die linke Seite von [Fig. 17](#) bzw. die linke Seite von [Fig. 18](#) zeigen, dass die Daten der Modelle M1 und M5 in den Ursprungsdaten die Eckenliste enthalten (**107** in [Fig. 17](#)). Diese Modelldaten der Ursprungsdaten erzeugt ein dreidimensionaler Modellierer, wobei jedes Modell eine Modellmatrix hat (**109** in [Fig. 17](#)), ein Modell, das ein Polygon hat, eine Eckenliste (**107** in [Fig. 17](#)) und eine Polygonliste (**106** in [Fig. 17](#)) besitzt, und ein Modell, das die Ecken beeinflusst, eine Gewichtungsliste aufweist (**108** in [Fig. 17](#)). Die ursprünglichen Eckenlisten auf der linken Seite von [Fig. 18](#) entsprechen den Eckenlisten der Modelle M1 und M5, die die oben erwähnten Modelle sind, die Polygone besitzen. Damit sind die hier dargestellten Ecken tatsächliche Ecken, und die Eckenliste umfasst mindestens die Positionskoordinatendaten und die Normalendaten der Ecken.

[0113] Zuerst ermittelt der Konvertierer die globalen Koordinaten der Positionskoordinaten und Normalendaten der tatsächlichen Ecken des integrierten Formgebungsmodells in der Grundform (S62). Diese Operation wird beispielsweise dadurch ausgeführt, dass die tatsächlichen Eckendaten der jeweiligen Modelle mit der Modellmatrix der Modelle M1 und M5 multipliziert werden. Dadurch werden die Daten über die tatsächlichen Ecken V101–V113 und V501–V506 in [Fig. 4](#) in das globale Koordinatensystem umge-

setzt.

[0114] Nun werden die tatsächlichen Ecken der Ursprungsmodelldaten in vier Gruppen eingeteilt, und zwar abhängig von der Gewichtung anhand der Abarbeitungsfolge der Modellhierarchie (S62, siehe die Schritte 121–124 in [Fig. 17](#)). In [Fig. 18](#) wird die Modellhierarchie beispielsweise in der Folge M1, M2, M3, M5 und M4 abgearbeitet. Die Ecke V101 des Modells M1 ist eine Ecke, die zu 100 Prozent von der Position des Modells M1 abhängt und zur Gruppe der eigenen Ecken gehört, die keine Gewichtung aufweist und keine Gewichtungsberechnung erfordert. Die Ecke V102 des Modells M1 ist in der Gewichtungstabelle in [Fig. 6](#) eingetragen und wird von den Modellen M2, M3 und M4 mit 30 Prozent, 30 Prozent und 40 Prozent beeinflusst, siehe [Fig. 8](#). Damit wird die Ecke V102 (M2), die von dem Modell M2 beeinflusst wird, in die Anfangsgewichtsgruppe eingeordnet, die Ecke V102 (M3), die von dem Modell M3 beeinflusst wird, wird in die Mittelgewichtsgruppe eingeordnet, und die Ecke V102 (M4), die von dem Modell M4 beeinflusst wird, wird in die Endgewichtsgruppe eingeordnet.

[0115] Für die Ecke V103 des Modells M1 wird ebenfalls die Ecke V103 (M2), die von dem Modell M2 beeinflusst wird, in die Anfangsgewichtsgruppe eingeordnet, die Ecke V103 (M3), die von dem Modell M3 beeinflusst wird, wird in die Mittelgewichtsgruppe eingeordnet, und die Ecke V103 (M4), die von dem Modell M4 beeinflusst wird, wird in die Endgewichtsgruppe eingeordnet. Für die Ecke V104 des Modells M1 wird die Ecke V104 (M2), die von dem Modell M2 beeinflusst wird, in die Anfangsgewichtsgruppe eingeordnet, und die Ecke V104 (M3), die von dem Modell M3 beeinflusst wird, wird in die Endgewichtsgruppe eingeordnet. Wie [Fig. 18](#) zeigt, werden die Ecken V105 und V108–V112 ebenfalls in die Anfangsgewichtsgruppe und in die Endgewichtsgruppe eingeordnet.

[0116] [Fig. 18](#) zeigt, dass die Ecke V106 des Modells M1, die zu 100 Prozent vom Modell M3 beeinflusst wird, in die eigene Gruppe eingeordnet wird. Die Ecke V107 wird zu 100 Prozent vom Modell M1 beeinflusst, und die Ecke V113 wird zu 100 Prozent vom Modell M4 beeinflusst. Daher werden die Ecke V107 (M1) und die Ecke V113 (M4) in die eigene Gruppe eingeordnet.

[0117] Die tatsächlichen Ecken V501–V506 des Modells M5 hängen alle von der Position des Modells M5 ab und erfahren keine Gewichtung durch andere Modelle. Daher werden die Ecken V501 (M5)–V506 (M5) jeweils in die eigene Gruppe eingeordnet und auf das Modell M5 verteilt. Auf diese Weise werden die tatsächlichen Ecken in vier Eckengruppen eingesortiert. Nun sortiert der Konvertierer die Ecken in den vier obigen Gruppen für jedes Modell, das diese

Ecken beeinflusst (S66, Schritte 125–128 in [Fig. 17](#)). Im Fall des Beispiels in [Fig. 18](#) werden beispielsweise die vier Ecken V101 (M1), V106 (M3), V107 (M1) und V113 (M4), die zur eigenen Gruppe des Modells M1 gehören, für jedes der Modelle M1, M3 und M4 sortiert, die diese Ecken beeinflussen, und beispielsweise in der Folge V101 (M1), V107 (M1), V106 (M3) und V113 (M4) neu angeordnet. Die neun Ecken V102 (M2)–V105 (M2) und V108 (M2)–V112 (M2), die zur Anfangsgewichtsgruppe des Modells M1 gehören, verbleiben in dieser Reihenfolge, da diese Ecken alle vom Modell M2 beeinflusst werden. Die beiden Ecken V102 (M3) und V103 (M3), die zur Mittelgewichtsgruppe des Modells M1 gehören, verbleiben in dieser Reihenfolge. Die neun Ecken V102 (M4), V103 (M4), V104 (M3), V105 (M3) und V108 (M3)–V112 (M3), die zur Endgewichtsgruppe gehören, werden für jedes der Modelle M3 und M4 umgeordnet, das diese Ecken beeinflusst. Die sechs Ecken des Modells M5 verbleiben in der gleichen Reihenfolge, da diese Ecken alle vom Modell M5 beeinflusst werden.

[0118] Nun erzeugt der Konvertierer das Format des gemeinsamen Eckenpuffers für die Ecken, die zur eigenen Gruppe und zur Anfangsgewichtsgruppe gehören, aus den obigen vier Gruppen (S68, Schritt 129 in [Fig. 17](#)). Dieser gemeinsame Eckenpuffer ist in der Tat ein Puffer zum Speichern der tatsächlichen Eckenliste, in dem zuerst die tatsächlichen Ecken für jedes Modell gruppiert werden, zu dem die Ecken gehören. Anschließend werden die Ecken für jede eigene Gruppe und Anfangsgewichtsgruppe gruppiert, und zuletzt werden die Ecken für jedes Modell gruppiert, das diese Ecken beeinflusst.

[0119] Im Fall des Beispiels in [Fig. 18](#) hat die Liste im gemeinsamen Eckenpuffer für das Modell M1 Ecken V101 und V107 für das einflussnehmende Modell M1, die Ecke V106 für das Modell M3 und die Ecke V113 des Modells M4 in der eigenen Gruppe. In der Anfangsgewichtsgruppe hat die Liste neun Ecken V102–V112 für das einflussnehmende Modell M2. In der Anfangsgewichtsgruppe sind keine Ecken für die Modelle M1, M3 und M4 vorhanden.

[0120] Für neue Eckenkennungen werden den im gemeinsamen Eckenpuffer eingetragenen Ecken neue Reihenfolgennummern vx0–vx12 zugewiesen. Zum Zuweisen einer Eckenkennung, siehe die Beschreibung im Schritt 129 in [Fig. 17](#), kann man eine Reihenfolgennummer, die mit einem Versatzwert beginnt, der aus der Anzahl der Ecken eines jeden Modells im gemeinsamen Eckenpuffer besteht, als Eintragsnummer verwenden. Für den Fall des Modells M1 wird beispielsweise die erste Adresse zu ADD (M1), und für das Gebiet der Ecke V102 wird ADD (M1) + 4 als Eckenkennung eingestellt, die anzeigt, dass die Ecke V102 die fünfte Eintragsnummer nach der ersten Adresse ADD (M1) ist. Da im Modell M1 13

Ecken vorhanden sind, ist der Versatzwert des Modells M1 13. Daher wird der Versatzwert 13 des Modells M1 für die erste Adresse ADD (M5) des folgenden Modells; des Modells M5, verwendet, und den gemeinsamen Ecken des Modells M5 werden Eintragsnummern 0–6 zugewiesen, die mit der ersten Adresse ADD (5) beginnen.

[0121] Verwendet man einen Versatzwert und eine Eintragsnummer in dieser Weise, kann man leicht eine Adresse einstellen, die ein Gebiet für jede Ecke im Puffer innerhalb des Speicherbereichs anzeigt, wenn die gemeinsame Eckenliste für jedes Modell im Puffer gespeichert ist. Dieser Gegenstand wird später ausführlicher erklärt. Ein derartiger Versatzwert und eine solche Eintragsnummer werden zur Formatinformation des gemeinsamen Eckenpuffers.

[0122] Die erwähnte gemeinsame Eckenliste ist in der Tat die gleiche Liste wie die tatsächliche Eckenliste eines jeden Modells. Durch das Speichern dieser Liste im Speicher und das Erzeugen des gemeinsamen Eckenpuffers kann man die Integration (Addition) des multiplizierten Werts der Gewichtungswerte und der Koordinaten- oder Normalendaten in der Gewichtungsoperation für den gemeinsamen Eckenpuffer ausführen. Durch das Zuweisen einer neuen Eckenkennung für jede Ecke und das Verwenden dieser Eckenkennung in der weiter unten erwähnten Eckenliste und Polygonliste können die Bezugskennungen der Ecken vereinfacht werden, die während der Arbeit der Zeichnungsbibliothek verwendet werden.

[0123] Da neue Eckenkennungen zugewiesen werden, werden die Eckenkennungen in der Polygonliste, die jedes Modell besitzt, ebenfalls auf die neuen Eckenkennungen konvertiert (S68). Durch diese Umwandlung kann der Zugriff auf Daten im gemeinsamen Eckenpuffer vereinfacht werden, wenn das Zeichnen (Rendern) der Polygone nach dem Ende des Gewichtungsvorgangs der Modelle ausgeführt wird, und die Geschwindigkeit des Zeichnungsabschritts (Renderschritts) durch die Zeichnungsbibliothek, bei dem eine Echtzeitverarbeitung gefordert ist, kann erhöht werden.

[0124] Wie in [Fig. 8](#) erklärt muss jede erzeugte Ecke auf die Modelle verteilt werden, die die Ecke beeinflussen. Daher verteilt der Konvertierer die vier Eckenarten auf die Modelle, die jeweils die Ecken beeinflussen (S70). Verteilen bedeutet hier das Aufführen der Ecke in der Eckenliste des Modells, das die Ecke beeinflusst, und das Konvertieren der Koordinatendaten und der Normalendaten der Ecke in das lokale Koordinatensystem des Modells. Das Umwandeln in das lokale Koordinatensystem erfolgt durch das Multiplizieren der Daten einer jeden Ecke im globalen Koordinatensystem, die im Schritt S62 bestimmt werden, mit der inversen Matrix der Modell-

matrix, die die Position des Modells angibt.

[0125] Der Konvertierer richtet die erwähnten verteilten Ecken in der Eckenliste des Verteilungszielmodells aus (S72). Diese Eckenliste wird für jede der vier Gruppen erzeugt. Für die Referenzkennungen der ausgerichteten Ecken werden neue Eckenkennungen (z. B. ein Versatzwert und eine Eintragsnummer) gemäß dem Format des gemeinsamen Eckenpuffers verwendet. Dadurch wird der Zugriff auf einen Bereich des gemeinsamen Eckenpuffers erleichtert, in dem ein Verarbeitungsergebnis gespeichert ist, wenn die Gewichtungsoperation für eine Ecke in der Eckenliste vorgenommen wird.

[0126] Für die Beispiele in [Fig. 18](#) und [Fig. 19](#) werden die Ecken in vier Gruppen in [Fig. 18](#) auf die Eckenlisten der Modelle verteilt, die die Ecken beeinflussen, d. h. auf die Modelle, die in den Zeichnungen in Klammern dargestellt sind (z. B. das Modell M1 im Fall von V101 (M1)). Dadurch wird die in [Fig. 19](#) dargestellte Eckenliste für jedes Modell erzeugt. In der Eckenliste des Modells M1 werden beispielsweise die Koordinatendaten VERT und die Normalendaten NORM der Ecken V101 (M1) und V107 (M1) angeordnet, die zur eigenen Gruppe gehören. Die jeweilige Referenzkennung einer Ecke wird mit "Ecken-ID (vx0, 2)" bezeichnet. D. h., "Ecken-ID (vx0, 2)" bezeichnet zwei Eckenbereiche in dem Gebiet, in dem die Eckenkennung im gemeinsamen Eckenpuffer in [Fig. 18](#) den Wert vx0 hat. Eine andere Art, Referenzkennungen zuzuweisen, besteht beispielsweise darin, den Eckendaten in der Eckenliste neue Eckenkennungen vx0 und vx1 als zugeordnete Daten zuzuweisen.

[0127] Wie das Beispiel in [Fig. 19](#) zeigt, hat die Eckenliste des Modells M2 keine Ecken, die zur eigenen Gruppe gehören. Sie besitzt neun Ecken, die zur Anfangsgewichtsgruppe gehören. Im Fall einer Ecke, die zur Anfangsgewichtsgruppe gehört, enthalten die Eckendaten Gewichtungsdaten NFlagsW, die den Gewichtungswert WT und die Kennungen der zu speichernden tatsächlichen Ecken enthalten (vx4 und vx5 für das Beispiel in [Fig. 19](#)), und zwar zusätzlich zu den Koordinatendaten VERT und den Normalendaten NORM. Diese tatsächliche Eckenkennung ist durch den Versatzwert eines jeden Eckenblocks des gemeinsamen Eckenpuffers und die Eintragsnummer dieses Blocks bestimmt, siehe [Fig. 13](#). Daher kann man in den Gewichtungsdaten NFlagsW den Eintragswert als Eckenkennung verwenden.

[0128] Die Eckenliste des Modells M3 in [Fig. 19](#) enthält Eckendaten über die Ecke V106 (M3), die zu der eigenen Gruppe gehört, Eckendaten (Koordinatendaten, Normalendaten und Gewichtungsdaten) zu den Ecken V102 (M3) und V103 (M3), die zur Mittelgewichtsgruppe gehören, und Eckendaten (Koordinatendaten, Normalendaten und Gewichtungsdaten)

zu den sieben Ecken, die zur Endgewichtsgruppe gehören. Die Eckenliste des Modells M4 enthält eine Ecke V113 (M4), die zur eigenen Gruppe gehört, und zwei Ecken V102 (M4) und V103 (M4), die zur Endgewichtsgruppe gehören.

[0129] Werden die genannten Schritte S62–S72 für alle Modelle ausgeführt, so werden die Polygonliste **116**, die auf den neuen Eckenkennungen beruht, und die Eckenliste **117**, die in vier Gruppen unterteilt ist, als jeweilige Modelldaten **115** erzeugt, siehe [Fig. 17](#). Ein Beispiel ist in [Fig. 19](#) dargestellt. Es wird auch die in [Fig. 13](#) dargestellte gemeinsame Eckenliste erzeugt, und es wird Formatinformation (z. B. Versatzwert und Eintragsnummer) des zugehörigen gemeinsamen Eckenpuffers erzeugt. Dadurch kann die Zeichnungsbibliothek den gemeinsamen Eckenpuffer erzeugen und die Daten über die tatsächlichen Ecken speichern. Sie kann die Bearbeitung der eigenen Ecken, die Bearbeitung der Anfangsgewichtsecken, die Bearbeitung der Mittelgewichtsecken und die Bearbeitung der Endgewichtsecken nacheinander gemäß der Eckenliste ausführen, die auf jedes Modell verteilt ist. Sie kann auch durch Zugriff auf die Polygonliste Polygone zeichnen, und zwar nach dem Gewichtungsvorgang abhängig von den neuen Eckenkennungen und unter Verwendung der Ecken- daten, die in den gemeinsamen Eckenpuffer geschrieben sind.

[0130] Zusätzlich zum Erzeugen der Formatinformation des gemeinsamen Eckenpuffers und der Eckenliste und der Polygonliste für jedes Modell, siehe [Fig. 14](#), simuliert der Konvertierer die Zeichnungszeit (Renderzeit) eines jeden Modells und fügt einen Zeichnungsbefehl oder Nicht-Zeichnungsbefehl in die Polygonliste ein. Hierfür erfasst der Konvertierer den Zeitpunkt, zu dem die Gewichtungsberechnung für die Ecken endet, die die tatsächlichen Ecken im Modell beeinflussen.

[0131] Zum Erfassen des Zeitpunkts stellt der Konvertierer den Zählerwert fest, der angibt, wie oft die Gewichtungsberechnung für jedes Modell vorgenommen wird, das Polygone besitzt (**S36**). Dieser Zählerwert ist gleich der Anzahl der Ecken, die in jedem Modell in vier Gruppen unterteilt sind. D. h., der Zählerwert für das Modell M1 beträgt 24, und der Zählerwert für das Modell M5 beträgt 6. Anders formuliert gibt dieser Zählerwert an, wie oft die Zeichnungsbibliothek das Schreiben oder Integrieren (Addieren) in den gemeinsamen Eckenpuffer ausführt. Das Ende der Gewichtungsberechnung nach dem Stand des Zählerwerts in jedem Modell bedeutet, dass die Gewichtungsberechnungen von Modellen, die die Ecken beeinflussen, zu Ende sind. Es bedeutet auch, dass die Daten der tatsächlichen Ecken bestimmt sind, und dass die Vorbereitungen zum Zeichnen (Rendern) abgeschlossen sind.

[0132] In der Zeichnungssimulation durch die Zeichnungsbibliothek werden die Matrizenberechnung und die Gewichtungsberechnung nacheinander für die Eckendaten der Eckenliste eines jeden Modells ausgeführt. Das Schreiben bzw. die Integration erfolgt im Eckenbereich im gemeinsamen Eckenpuffer (**S42**), und der Zähler des Modells, zu dem die geschriebenen oder integrierten Ecken gehören, wird um +1 erhöht (**S44**). Für den Fall des Beispiels in [Fig. 19](#) wird der gemeinsame Pufferbereich des Modells M1 im Speicher gesichert. Für die Eckenliste des Modells M1 werden die Koordinatendaten und die Normalendaten an neuen Positionen dadurch bestimmt, dass die Koordinatendaten VERT und die Normalendaten NORM der Ecke V101 (M1) mit der Modellmatrix des Modells M1 multipliziert werden. Anschließend wird mit dem Gewichtungswert 1,0 (100 Prozent) multipliziert, und diese Daten werden in den Bereich vx0 im gemeinsamen Eckenpuffer geschrieben, siehe [Fig. 18](#). Diese Ecke V101 (M1) ist eine eigene Gruppe, die zu 100 Prozent vom Modell M1 beeinflusst wird; daher ist der Gewichtungswert 1,0, und die Daten bleiben auch dann unverändert, wenn mit dem Gewichtungswert multipliziert wird. Für die Ecken der eigenen Gruppe kann man die Daten in den gemeinsamen Eckenpuffer schreiben, ohne die Gewichtungsberechnung (Multiplikation) auszuführen. Die gleiche Berechnung erfolgt für die Ecke V107 (M1), und die ermittelten Daten werden in den Bereich vx1 im gemeinsamen Eckenpuffer geschrieben. Dadurch nimmt der Zählerstand des Modells M1 den Wert 2 an.

[0133] Die genannten Schritte S42 und S44 werden in der Reihenfolge der Eckenliste in jedem Modell wiederholt. Die Ausführung der Schritte S42 und S44 endet für alle Ecken eines Modells (**S46**). Ob die Gewichtungsberechnung für dieses Modell beendet ist, wird dadurch festgestellt, dass man prüft, ob der Zählerwert des Modells gleich der Einstellung ist (**S48**). Für den Fall des Modells M1 in [Fig. 19](#) ist beim Zählerwert 2 die Einstellung noch nicht erreicht, die den Wert 24 hat, und zwar auch dann noch nicht, wenn die Gewichtungsberechnung für die beiden Ecken V101 (M1) und V107 (M1) beendet ist. Daher kann das Modell M1 noch nicht gezeichnet werden. Im Fall eines herkömmlichen Modells, das aus zahlreichen Polygone besteht, werden die Positionen (Koordinaten) und Normalen der Ecken, die die Polygone bilden, einheitlich nur dann bestimmt, wenn die Position des Modells definiert ist. Anders gesagt sind die neuen Koordinatendaten und Normalendaten definiert, wenn die Modellmatrix des Modells ermittelt ist und die Modellmatrix mit den Koordinatendaten und Normalendaten multipliziert ist. Dies bedeutet, dass das Zeichnen (Rendern) des Modells an dieser Stelle ausgeführt werden kann. Das Zeichnen des Modells kann also erfolgen, wenn die Matrizenoperation zu Ende ist.

[0134] Für den Fall des integrierten Formgebungsmodells (Hülle), das von anderen Modellen beeinflusst wird, kann das Zeichnen des Modells nicht vorgenommen werden, bis alle Gewichtungsberechnungen für diese Einflüsse zu Ende sind.

[0135] Ist die Gewichtungsberechnung für ein Modell noch nicht vollendet, so wird gemäß der Erfindung ein Nicht-Zeichnungsbefehl DRAWCP (ADD) in der Polygonliste dieses Modells erzeugt, und er wird in die später erwähnte Nicht-Zeichnungstabelle in [Fig. 21](#) eingeordnet, damit der gemeinsame Eckenpufferbereich für das nächste Modell gesichert ist (S50). Die Attributdaten ADD des obigen Nicht-Zeichnungsbefehls bezeichnen eine Adresse der Polygonliste. Im Fall des Beispiels in [Fig. 19](#) wird der Nicht-Zeichnungsbefehl DRAWCP (ADD) an den Anfang der Polygonliste des Modells M1 geschrieben. Dadurch wird die Polygonliste des Modells M1 vervollständigt. DRAWCP ist eine Abkürzung für "DRAW Cash Polygon".

[0136] Ist dagegen die Gewichtungsberechnung für das Modell beendet, so wird ein Zeichnungsbefehl DRAWDP (ADD) in die Polygonliste des Modells eingefügt (S52). Der Bereich des gemeinsamen Eckenpuffers wird freigegeben. Anstatt den Zeichnungsbefehl zuzufügen kann man eine Regel aufstellen, nach der das Zeichnen (Rendern) des Modells unmittelbar erfolgt, wenn kein Nicht-Zeichnungsbefehl in die Polygonliste geschrieben wird. Anders formuliert kann das Zeichnen (Rendern) als Voreinstellung erfolgen, wenn die Matrizenberechnung und die Gewichtungsberechnung für die Eckenliste des Modells beendet ist. In diesem Fall wird der erwähnte Zeichnungsbefehl nicht in die Polygonliste des Modells eingefügt, und es wird entscheidend, ob der Nicht-Zeichnungsbefehl vorhanden ist oder nicht.

[0137] Ist die Berechnung für die Eckenliste des Modells beendet, so werden die Eckenliste und die Polygonliste des Modells ausgegeben (S54). Die Polygonliste enthält die Anordnungskennung der Nicht-Zeichnungstabelle für den Nicht-Zeichnungsbefehl und die Anordnungskennung der Zeichnungstabelle für den Zeichnungsbefehl.

[0138] Auch wenn die Verarbeitung für alle Ecken in der Eckenliste des Modells noch nicht zu Ende ist, muss das Ende der Gewichtungsberechnung des Nicht-Zeichnungsmodells durch Zugriff auf den Zählerwert für jedes Modell erkannt werden (S58). Ein Zeichnungsbefehl DRAWDP (ADD) wird in der Polygonliste des derzeit verarbeiteten Modells erzeugt, und zwar für das Nicht-Zeichnungsmodell, für das die Gewichtungsberechnung beendet ist. Dadurch wird der Bereich des gemeinsamen Eckenpuffers für das Modell freigegeben, für das das Zeichnen möglich geworden ist (S60). DRAWDP (ADD) ist eine Abkürzung für "DRAW Draw Polygon".

[0139] Die obige Simulationsverarbeitung wird für die Eckenliste aller Modelle ausgeführt.

[0140] Damit man die obigen Erklärungen am konkreten Fall verstehen kann, werden die Beispiele in [Fig. 18](#) und [Fig. 19](#) zur Erklärung herangezogen. [Fig. 20](#) zeigt ein Beispiel der Polygonliste, die für diesen Fall erzeugt wird. [Fig. 21](#) zeigt ein Beispiel der Nicht-Zeichnungstabelle. Zuerst führt der Konvertierer die Matrizenberechnung und die Gewichtungsberechnung für die Eckenliste des Modells M1 aus und schreibt das Ergebnis in die zugehörigen Bereiche im gemeinsamen Eckenpuffer. Da die Gewichtungsberechnung auch dann noch nicht beendet ist, wenn die Verarbeitung für alle Eckendaten in der Eckenliste zu Ende ist, wird der Nicht-Zeichnungsbefehl DRAWCP (0) am Anfang der Polygonliste **145** für das Modell M1 eingeordnet. Zu diesem Zeitpunkt wird der Zusammenhang zwischen der ersten Adresse ADD1 der Polygonliste und den Attributdaten "0" dieses Befehls in die Nicht-Zeichnungstabelle **143** eingeordnet.

[0141] Daraufhin führt der Konvertierer die Matrizenberechnung und die Gewichtungsberechnung für die Eckenliste des Modells M2 in der gleichen Weise aus und schreibt oder integriert (addiert) das Ergebnis in die zugehörigen Bereiche im gemeinsamen Eckenpuffer. Da die Gewichtungsberechnung für die Ecken des Modells M1 noch nicht beendet ist, wenn die Berechnung für alle Ecken der Eckenliste des Modells M2 endet, wird die gleiche Verarbeitung für das folgende Modell, das Modell M3, ausgeführt. Ist die Verarbeitung der Eckenliste des Modells M4 beendet, so ist die Gewichtungsberechnung für die tatsächlichen Ecken des Modells M1 beendet, und das Zeichnen des Modells M1 wird möglich. Damit wird der Zeichnungsbefehl DRAWDP (0) für das Modell M1, das sich bisher im Nicht-Zeichnungszustand befunden hat, am Anfang der Polygonliste **147** des Modells M4 eingeordnet.

[0142] In der Eckenliste des Modells M5, die in [Fig. 19](#) nicht dargestellt ist, gehören alle Ecken zur eigenen Gruppe. Damit befindet sich das Modell M5 in einem zum Zeichnen freigegebenen Zustand, nachdem die Matrizenberechnung und die Gewichtungsberechnung der Eckenliste des Modells M5 beendet sind. Daher wird der Zeichnungsbefehl DRAW für das Modell M5 in die Polygonliste **146** des Modells M5 eingeordnet. Wird das Zeichnen per Voreinstellung ausgeführt, wenn kein Nicht-Zeichnungsbefehl eingetragen ist, so braucht der Zeichnungsbefehl DRAW für das Modell M5 nicht eingetragen zu werden.

[0143] Jede Polygonliste **145**, **146** und **147** enthält Polygonnamen PG und Eckenkennungen der Ecken, die die Polygone bilden. Für die Eckenkennungen werden neue konsistente Eckenkennungen vx0 und vx1, die im gemeinsamen Eckenpuffer zugewiesen

wurden, für den Fall des Beispiels in [Fig. 20](#) verwendet. Die Polygonliste kann mit Hilfe einer aktuellen Adresse anhand der Eintragsnummer und des Versatzwerts erzeugt werden, falls der gemeinsame Eckenpuffer als Eckenkennung erzeugt wurde. Ein Beispiel hierfür wird im Weiteren erklärt.

[0144] In der Nicht-Zeichnungstabelle **143** in [Fig. 21](#) entspricht die erste Adresse ADD1 der Polygonliste des Modells der Kennung ID = 0 des Modells M1. Die Nicht-Zeichnungstabelle **143** ist nicht immer erforderlich, und man kann die erste Adresse ADD1 der Polygonliste direkt den Attributdaten des Nicht-Zeichnungsbefehls DRAWCP zuweisen.

[0145] Wie oben erklärt erzeugt der Konvertierer Konvertierungsdaten (nicht in Echtzeit), die die Formatinformation des gemeinsamen Eckenpuffers und der Eckenliste und Polygonliste eines jeden Modells enthalten, aus den Ursprungsdaten des integrierten Formgebungsmodells, die ein dreidimensionaler Modellierer erzeugt hat. Diese Konvertierungsdaten werden in einem Aufzeichnungsmedium für Spiele, beispielsweise einer CD-ROM, zusammen mit dem Spielprogramm und der Zeichnungsbibliothek aufgezeichnet.

[0146] Nun werden die Eckenkennungen beim Erzeugen und Freigeben des gemeinsamen Eckenpuffers erklärt. Die Zeichnung in [Fig. 22](#) dient der Erklärung der Eckenkennungen. Im Beispiel in [Fig. 22](#) wird das Modell M6, das ein Polygon aufweist, das im Folgenden zu verarbeiten ist, den genannten Modellen M1–M5 zugefügt, damit die Eckenkennungen leicht erklärt werden können. [Fig. 22A](#) zeigt einen Status, in dem die Ecken des Modells in die eigene Gruppe, die Anfangsgewichtsgruppe, die Mittelgewichtsgruppe und die Endgewichtsgruppe eingesortiert werden. Das Format des Eckenpuffers wird aus den Ecken erzeugt, die zu der eigenen Gruppe und der Anfangsgewichtsgruppe gehören. In diesem Status wird allen Ecken eine Reihenfolgennummer vx0 – vx(h + i + j – 1) zugewiesen. Dabei sind h, i und j jeweils die Versatzwerte der Modelle M1, M5 und M6. Dabei wird unterstellt, dass der Konvertierer in diesem Status eine Simulationsoperation ausgeführt hat, durch die die gemeinsamen Eckenpufferbereiche erzeugt und freigegeben wurden, siehe [Fig. 22B](#).

[0147] Von den obigen Modellen M1–M5 weisen das Modell M1 und das Modell M5 tatsächliche Ecken auf. Die Verarbeitung der Eckenliste wird in der Reihenfolge der Modelle M1, M2, M3, M5 und M4 vorgenommen, die der Verarbeitungsabfolge der Modellhierarchie entspricht. Ist die erste Adresse des gemeinsamen Eckenpufferbereichs ADDm, so wird der gemeinsame Eckenpufferbereich des Modells M1 zuerst an der Adresse ADDm oder weiter unten erzeugt ([Fig. 22B](#) (B-1)). Daraufhin wird der gemeinsame Eckenpufferbereich des Modells M5 nach dem

Bereich des Modells M1 erzeugt ([Fig. 22B](#) (B-2)). An dieser Stelle wird die Adresse der Ecke des Modells M1 durch die erste Adresse ADDm und den Eintragswert 0 – (h – 1) spezifiziert. Die Adresse einer Ecke des Modells M5 wird durch die erste Adresse ADDm und den Versatzwert h des Modells M1 und den Eintragswert 0 – i des Modells M5 festgelegt. Daher wird an dieser Stelle die Adresse einer gemeinsamen Ecke der Modelle M1 und M5 durch eine Reihenfolgennummer 0 – (h + i – 1) bestimmt. Diese Adresse wird für die Eckenkennung verwendet.

[0148] Anders ausgedrückt werden die Modelle M1 und M5 an dieser Stelle gezeichnet, und anschließend werden die gemeinsamen Eckenpufferbereiche der Modelle M1 und M5 freigegeben. Daher kann man diese Adressen für alle Eckenkennungen der Polygonlisten der Modelle M1 und M5 verwenden. Dadurch steigt die Wirksamkeit der Verarbeitung durch die Zeichnungsbibliothek.

[0149] Ist das Zeichnen der Modelle M1 und M5 beendet, so werden die gemeinsamen Eckenpufferbereiche der Modelle M1 und M5 nicht mehr benötigt. Daher wird der gemeinsame Eckenpufferbereich des Modells M6, das als nächstes gezeichnet wird, wieder an der Eintragsnummer 0 – (j – 1) nach der ersten Adresse ADDm erzeugt. Diese Adressen werden für die Eckenkennungen der Eckenliste und der Polygonliste des Modells M6 verwendet. Anders formuliert überlappen sich die Adressen der gemeinsamen Eckenpufferbereiche der verschiedenen Modelle, damit so wenig gemeinsame Eckenpufferbereiche wie möglich erzeugt und unterhalten werden müssen. Diese Adressenüberlappung führt nicht zu Problemen, da auf die Eckenliste und die Polygonliste in der Verarbeitungsfolge der hierarchischen Struktur des Modells zugegriffen wird.

[0150] Eine abgewandelte Form besteht darin, eine Entsprechungstabelle zwischen den Adressen der gemeinsamen Eckenpufferbereiche und den konsistenten Eckenkennungen in einem Simulationsvorgang des Konvertierers mit Hilfe der konsistenten Eckenkennungen vx0 – vx(h + i + j – 1) für die Eckenliste und die Polygonliste zu erzeugen. In diesem Fall kann die Zeichnungsbibliothek die Adressen der Pufferbereiche der Eckenkennungen, die der Eckenliste und der Polygonliste entsprechen, durch einen Zugriff auf die Entsprechungstabelle feststellen.

[0151] Die Zeichnung in [Fig. 23](#) zeigt ein weiteres Beispiel einer Eckenliste. Dieses Beispiel gilt für den Fall, dass die Adressen des gemeinsamen Eckenpufferbereiches als Eckenkennungen verwendet werden. Die Eckenliste **200** besitzt einen Satz ("dicken Brocken") von Eckendaten für die vier Gruppen zwischen dem Anfangsdeskriptor **201** und dem Enddeskriptor **203** der Eckenliste. **202** ist der Deskriptor, der das Ende des Eckendatensatzes bezeichnet. An-

ders ausgedrückt weist die Eckenliste **200** einen Satz **218** an eigenen Ecken ohne Gewichtung auf, einen Satz **228** der Ecken, die zur Anfangsgewichtsgruppe gehören, einen Satz **238** der Ecken, die zur Mittelgewichtsgruppe gehören, und einen Satz **248** der Ecken, die zur Endgewichtsgruppe gehören. Jeder Satz besitzt einen Deskriptor **211**, der den Datensatznamen der Eckengruppe angibt, die keine Gewichtung aufweist, und einen Deskriptor **221**, der den Datensatznamen der Eckengruppe angibt, die eine Gewichtung hat. Diese Deskriptoren enthalten ein Flag **212**, das anzeigen, dass die Gewichtungsberechnung noch andauert, eine Satzgröße **213** und eine Klassifizierung von Anfang, Mitte und Ende **222**, **232** und **242**.

[0152] Jeder Satz enthält auch Deskriptoren, die den Versatzwert **214** des Eckenlistenblocks des gemeinsamen Eckenpuffers und die Anzahl von Eckendaten **215** in diesem Satz bezeichnen. Die Eckendaten umfassen die Koordinatendaten **216** (VERT), die Normalendaten **217** (NORM) und die Gewichtungsdaten (NFlagsW) wie im in [Fig. 19](#) dargestellten Fall. Die Gewichtungsdaten (NFlagsW) haben die Eintragsnummer **226** der Ecke im gemeinsamen Eckenpuffer und den Gewichtungswert (%) **227**. Den Bereich (Eckenkennung) einer tatsächlichen Ecke im gemeinsamen Eckenpuffer kann man durch einen Versatzwert **214** und die Eintragsnummer **226** bestimmen.

[0153] Nun wird eine Prozedur der Zeichnungsbibliothek erklärt. [Fig. 24](#) zeigt ein Flussdiagramm einer Prozedur der Zeichnungsbibliothek. Wie in [Fig. 13](#) erzeugt das Spielprogramm die Matrix zum Bewegen der Modelle, aus denen Figuren bestehen, abhängig von den Steuerdaten, die die Bedienperson im Verlauf des Spielfortschritts eingibt (S22). Daraufhin zeichnet die Zeichnungsbibliothek die Polygone (S24).

[0154] Die Prozedur der Zeichnungsbibliothek gleicht dem Simulationsvorgang des Konvertierers. Die Zeichnungsbibliothek zeichnet mit Hilfe der Konvertierungsdaten in Echtzeit Polygone. Die Zeichnungsbibliothek ermittelt die Koordinatendaten und Normalendaten der Ecken nach der Bewegung durch die Matrizenberechnung in der Abfolge der Eckenliste eines jeden Modells gemäß der Verarbeitungsfolge der hierarchischen Struktur der Modelle (S80). Die Koordinatendaten und Normalendaten werden durch das Multiplizieren der Konvertierungsdaten mit der Modellmatrix ermittelt und anschließend in das globale Koordinatensystem umgewandelt (S80). Im Zuge dieser Matrizenberechnung wird der gemeinsame Eckenpufferbereich des Modells im Speicher gesichert (S78). Ein derartiger gemeinsamer Eckenpufferbereich kann je nach Bedarf abhängig von der Formatinformation in den Konvertierungsdaten gesichert werden. Ist die Ecke, für die die Matrizenberechnung

ausgeführt wird, eine Ecke, die in die eigene Gruppe eingesortiert ist (S82), so wird in der Gewichtungsberechnung einfach mit 100 Prozent (1,0) multipliziert, d. h. tatsächlich werden die durch die Matrizenoperation gewonnenen Koordinatendaten und Normalendaten im zugeordneten Bereich des gemeinsamen Eckenpuffers aufgezeichnet (S84).

[0155] Ist die Matrix, für die die Matrizenoperation vorgenommen wird, eine gewichtete Ecke, d. h. eine Ecke, die zur Anfangsgewichtsgruppe, zur Mittelgewichtsgruppe oder zur Endgewichtsgruppe gehört (S86), so erfolgt die Gewichtungsberechnung mit Hilfe der Gewichtungsdaten. Die Koordinatendaten und Normalendaten im Berechnungsergebnis werden in den entsprechenden gemeinsamen Eckenpuffer geschrieben oder integriert (addiert). Für eine Ecke, die zur Anfangsgewichtsgruppe gehört, d. h. eine zuerst auftretende Ecke, wird das Berechnungsergebnis in den entsprechenden gemeinsamen Eckenpuffer geschrieben. Für eine Ecke, die zur Mittelgewichtsgruppe oder zur Endgewichtsgruppe gehört, wird das Berechnungsergebnis in den bereits eingetragenen Wert integriert (zu diesem addiert). Der Zähler des Modells, bei dem das Berechnungsergebnis aufgezeichnet wird, wird um +1 inkrementiert (S90).

[0156] Die angegebene Matrizenberechnung und Gewichtungsberechnung wird für alle Eckendaten in der Eckenliste des Modells ausgeführt (S92). Da für alle Eckenkennungen konsistente Kennungen verwendet werden und die Eckenkennungen den Referenzindizes des Eckendatenbereichs des gemeinsamen Eckenpuffers entsprechen, kann man diese Gewichtungsberechnung mit sehr hoher Geschwindigkeit ausführen. Die Eckenliste wird auch für jedes Modell unterteilt. Damit kann man die gleiche Modellmatrix wiederholt für die Ecken im gleichen Modell in der Matrizenberechnung verwenden. D. h., dass es nicht erforderlich ist, die Modellmatrix häufig zu verändern, die die Zeichnungsbibliothek während der Matrizenberechnung in der CPU einstellen muss. Dies erhöht die Wirksamkeit der Matrizenberechnung und ermöglicht das Zeichnen in Echtzeit.

[0157] Die Eckenliste wird jeder Eckengruppe zugeordnet, so dass sich die Gewichtungsberechnungsprozedur für eine Eckengruppe vereinheitlichen lässt. Anders gesagt wird für die Ecken, die zur eigenen Gruppe gehören, das Matrizenberechnungsergebnis direkt in den gemeinsamen Eckenpuffer geschrieben. Für die Ecken, die zur Anfangsgewichtsgruppe gehören, wird das Ergebnis der Matrizenberechnung und der Gewichtungsberechnung in den gemeinsamen Eckenpuffer geschrieben. Für die Ecken, die zur Mittelgewichtsgruppe und zur Endgewichtsgruppe gehören, wird das Ergebnis der Matrizenberechnung und der Gewichtungsberechnung in die Daten im gemeinsamen Eckenpuffer integriert (zu diesen addiert). Da sich die Prozedur auf diese Weise verein-

heitlichen lässt, wächst die Arbeitseffizienz der Zeichnungsbibliothek, wodurch ein Echtzeitbetrieb möglich wird.

[0158] Ist die Verarbeitung der Eckenliste eines Modells beendet, so greift die Zeichnungsbibliothek auf die Polygonliste dieses Modells zu. Ist ein Nicht-Zeichnungsbefehl DRAWCP in die Polygonliste (S94) geschrieben, so kann das Zeichnen (Rendern) des Modells nicht erfolgen, und das Modell wird in die Nicht-Zeichnungstabelle eingeordnet (S96). Ist ein Zeichnungsbefehl DRAWDP für ein Nicht-Zeichnungsmo~~d~~del in die Polygonliste (S98) geschrieben, so wird auf die Adresse des Nicht-Zeichnungsmo~~d~~del zugegriffen, das in die Nicht-Zeichnungstabelle eingeordnet ist, und das Zeichnen (Rendern) der Polygone wird in der Reihenfolge der Polygonliste vorgenommen (S100). Dieses Rendern der Polygone erfolgt mit Hilfe der Koordinatendaten und Normalendaten im globalen Koordinatensystem anhand der im gemeinsamen Eckenpuffer gespeicherten Verarbeitungsergebnisse. Ist das Zeichnen des Modells beendet, so ist der zum Modell gehörende Bereich des gemeinsamen Eckenpuffers nicht mehr erforderlich und wird freigegeben. Freigeben bedeutet hier, dass der Status des Bereichs so gesetzt wird, dass er bei Bedarf als weiterer gemeinsamer Eckenpuffer zugewiesen werden kann. Ist ein Zeichnungsbefehl DRAW, der das Zeichnen des Modells anordnet, in die Polygonliste eingetragen, so wird dieses Modell gezeichnet (gerendert), und der zugehörige gemeinsame Eckenpuffer wird freigegeben (S102).

[0159] Es ist auch möglich, die Polygone anhand der Polygonliste des Modells automatisch zu zeichnen, nachdem die Verarbeitung der Eckenliste im Modell beendet ist. Dies ist die voreingestellte Prozedur solange noch kein Nicht-Zeichnungsbefehl vorhanden ist. Vor den Zeichnen (Rendern) erfolgt falls erforderlich eine Normierung für die Normalendaten, für die die Eckenberechnung endet (S93). Die Normalendaten werden zum Ermitteln des Skalarprodukts ($\cos\theta$, θ ist der Winkel zwischen den Vektoren) zwischen dem Vektor des Lichts von der Lichtquelle und dem normierten Normalenvektor im Rendeschritt verwendet. Anhand des Skalarprodukts werden beispielsweise Verarbeitungen für diffuses Licht in Pixeln der Polygone vorgenommen. Damit ist abhängig von der Verarbeitung im Rendervorgang eine Normierung von Normalenvektoren (Vorgang, durch den der Vektor den Betrag 1 erhält) vorab erforderlich.

[0160] Im Zeichnungsschritt (Rendern) durch die Zeichnungsbibliothek ist es nicht erforderlich, den gemeinsamen Eckenpufferbereich für alle Modelle im Speicher abzulegen. Der zu einem Modell gehörende Eckenpufferbereich muss gespeichert werden, bis der nötige Gewichtungsvorgang beendet ist. Sind alle Gewichtungsvorgänge beendet und kann das

Modell gezeichnet werden, so kann das Zeichnen (Rendern) der Polygone des Modells gemäß dem Zeichnungsbefehl ausgeführt werden, der an den Anfang der Polygonliste eingeordnet ist. Ist das Zeichnen beendet, so werden die Eckendaten in diesem gemeinsamen Eckenpufferbereich nicht mehr benötigt, und dieser Bereich wird freigegeben, damit er für den gemeinsamen Eckenpufferbereich eines weiteren Modells verfügbar ist. Dies bedeutet, dass die Kapazität des gemeinsamen Eckenpufferbereichs im Speicher relativ klein sein kann. Dies ist ein großer Vorteil für eine Spieldatenmaschine, in der nur ein Speicher mit relativ geringer Kapazität verwendbar ist.

[0161] Auf diese Weise erzeugt der Konvertierer die Konvertierungstabelle im voraus. Dadurch kann die Zeichnungsbibliothek, die den Rendervorgang in Echtzeit ausführen muss, wie erwähnt die Matrizenberechnung und die Gewichtungsberechnung sehr wirksam ausführen. Man kann auch die Kapazität des Datenbereichs (gemeinsamer Eckenpufferbereich) der tatsächlichen Ecken, die im Speicher abgelegt werden müssen, verringern. Damit kann die Zeichnungsbibliothek Bilder von Figuren in Echtzeit erzeugen und die Spieldatenmaschine verwenden, die auf die Betätigung durch die Bedienperson reagiert.

[0162] In der obigen Ausführungsform wurde die Verarbeitung von Konvertierungsdaten durch den Konvertierer und die Verarbeitung durch die Zeichnungsbibliothek für mehrfach gewichtete Daten erklärt, wobei tatsächliche Ecken durch zahlreiche Modelle im integrierten Formgebungsmodell beeinflusst werden. Durch den Gebrauch eines solchen mehrfach gewichteten integrierten Formgebungsmodells kann man den Gelenkschnitt, der die Polygone verbindet, und die Form der Muskeln an der Oberfläche natürlicher zeichnen.

[0163] Abhängig von der Szene im Spiel ist jedoch eine komplizierte Eckenverarbeitung gefolgt von einer Mehrfachgewichtungs-Datenverarbeitung überflüssig, und zwar auch für eine Figur, die aus einem mehrfach gewichteten Formgebungsmodell besteht, wenn sich die Figur beispielsweise in einer entfernten Position von einem Betrachtungspunkt befindet. Dies ist so, weil eine Figur an einer entfernten Position von einem Betrachtungspunkt keine hochwertigen Bilder erfordert. Für eine Figur in Bildschirmmitte an einer Position in der Nähe des Betrachtungspunkts ist dies anders. In diesem Fall wird nicht die Eckenverarbeitung verwendet, der eine Mehrfachgewichtungsverarbeitung folgt, die Zeit und Hardwaredressourcen für die arithmetische Verarbeitung erfordert, sondern es wird die Eckenverarbeitung verwendet, der eine Einfachgewichtungsverarbeitung folgt. Damit kann man die Bildverarbeitungszeit für einen gesamten Bildschirminhalt und die Hardwaredressourcen so gering wie möglich halten. Einfachgewichtung bezieht sich auf ein Modell, das nur Ecken enthält, die ausschließ-

lich von einem Modell beeinflusst werden, d. h., das nur aus Ecken besteht, die zur eigenen Gruppe gehören.

[0164] [Fig. 25](#) zeigt ein Flussdiagramm mit der Prozedur, die der Konvertierer ausführt, um die Einfachgewichtungs-Eckenliste aus den Daten des mehrfach gewichteten integrierten Formgebungsmodells zu gewinnen. Dieses Flussdiagramm unterscheidet sich teilweise vom Flussdiagramm für die Aufteilung von Ecken auf Modelle und von der Eckenlistenerzeugung für mehrfach gewichtete Daten in [Fig. 16](#).

[0165] Das Konvertieren von Koordinaten und Normalendaten von tatsächlichen Ecken des integrierten Formgebungsmodells in der Grundform in das globale Koordinatensystem im Schritt S62 erfolgt ebenso wie bei der Mehrfachgewichtungsverarbeitung. Nun unterteilt der Konvertierer die tatsächlichen Ecken der Ursprungsmodelldaten entsprechend der Abarbeitungsfolge der Hierarchie der Modelle in gewichtungsabhängige Gruppen (S64). D. h., die tatsächlichen Ecken der Ursprungsmodelldaten werden in die eigene Gruppe und in die anderen Gruppen unterteilt, die für die Mehrfachgewichtung verwendet werden. Daraufhin konvertiert der Konvertierer die Ecken, die Gewichtungen (Einfluss) aufweisen, aus zahlreichen Modellen in die eigenen Ecken des Modells, deren Gewichtung am größten ist (S65). Im Fall des Beispiels in [Fig. 8](#) ist etwa an der Ecke V102, die Gewichtungen von drei Modellen M2, M3 und M4 hat, der Gewichtungswert vom Modell M4 am größten, nämlich 40 Prozent, so dass von den drei Ecken V102 (M2), V102 (M3) und V102 (M4) nur die Ecke V102 (M4) als Ecke gesetzt wird, die zum Modell M4 gehört. Die anderen Ecken V102 (M2) und V102 (M3) deren Gewichtungswert klein ist, werden nicht beachtet. Dadurch wird die Ecke V102 zu einer eigenen Ecke, die zu 100 Prozent vom Modell M4 beeinflusst wird.

[0166] Durch die Verarbeitung im Schritt S65 werden alle Ecken Einzelgewichtungsecken, die von einem Modell beeinflusst werden. Die weitere Verarbeitung unterscheidet sich nicht vom Fall der Mehrfachgewichtung in [Fig. 16](#). D. h., die Ecken werden für jedes Modell sortiert, das Einfluss auf die Ecken ausübt (S66), und allen Ecken werden konsistente Zahlen als neue Eckenkennungen zugewiesen, die ebenfalls in der Polygonliste eines jeden Modells verwendet werden (S68). Oder man kann wie im Fall der Mehrfachgewichtung eine Eckenkennung verwenden, die auf dem Versatzwert und der Eintragsnummer des gemeinsamen Eckenpuffers beruht.

[0167] Nun verteilt der Konvertierer alle Ecken auf die Modelle, die die Ecken beeinflussen (Modelle, zu denen die Ecken gehören), und er konvertiert die Eckendaten in das lokale Koordinatensystem des Modells (S70). Der Konvertierer ordnet diese verteilen

ten Ecken in die Eckenliste des Verteilungszielmodells ein (S72). Da in diesem Fall Einzelgewichtungen behandelt werden, treten in der Eckenliste eines jeden Modells anders als im Fall der Mehrfachgewichtung keine eigenen Gruppen oder Anfangsgewichtungsgruppen auf, und alle Ecken werden zu eigenen Ecken. Tatsächliche Ecken in der Eckenliste haben natürlich Eckenkennungen, die auf dem Format des gemeinsamen Eckenpuffers beruhen.

[0168] Anders ausgedrückt wird im Fall der Einzelgewichtung eine Liste der tatsächlichen Ecken, die die Polygone des Modells bilden, für jedes Modell im gemeinsamen Eckenpufferbereich erzeugt, und eine Liste der eigenen Ecken, die das Modell beeinflusst, wird in der Eckenliste eines jeden Modells erzeugt. Werden derartige Einzelgewichtungs-Konvertierungsdaten verwendet, so kann die Zeichnungsbibliothek nur eine Eckenoperation für eine Ecke ausführen. Dies vereinfacht den Zeichenvorgang und ermöglicht eine sehr schnelle Zeichnungsverarbeitung.

[0169] Die Einzelgewichtungs-Konvertierungsdaten erzeugt der Konvertierer bevorzugt für alle Mehrfachgewichtungs-Formatmodelle. Daher erzeugt der Konvertierer für die Mehrfachgewichtungs-Formatmodelle Mehrfachgewichtungs-Formatkonvertierungsdaten gemäß [Fig. 16](#) und Einzelgewichtungs-Formatkonvertierungsdaten nach [Fig. 25](#). Die Zeichnungsbibliothek führt die Zeichnungsverarbeitung mit Hilfe der Einzelgewichtungs-Formatkonvertierungsdaten während eines Spiels aus, wenn keine präzise Bewegung der Modelle erforderlich ist, beispielsweise wenn sich eine zu zeichnende Figur in einer entfernten Position von einem Betrachtungspunkt befindet oder wenn sich die Figur nicht in der Bildschirmmitte befindet oder wenn sich die Figur in einer Position hinter einem halbdurchsichtigen Polygon befindet. Die Zeichnungsbibliothek führt die Zeichnungsverarbeitung mit Hilfe der Mehrfachgewichtungs-Konvertierungsdaten nur für Figuren mit den am stärksten hervorstechenden Positionen in der Bildschirmmitte aus. Dadurch kann man die Last der Zeichnungsverarbeitung verringern, die für den ganzen Bildschirminhalt erforderlich ist.

[0170] Gemäß der Erfindung können der Zeichnungsbibliothek Modellkonvertierungsdaten mit einer optimalen Datenstruktur geliefert werden, die in einem Aufzeichnungsmedium für Spiele gespeichert sind (oder simuliert werden), und die Zeichnungsbibliothek führt die Bildverarbeitung in Echtzeit aus. Daher kann man integrierte Formgebungsmodelle, die eine natürlichere Bewegung von Gelenken und äußeren Flächen von Figuren implementieren können, in Echtzeit zeichnen.

[0171] Gemäß der Erfindung wird ein Bildverarbeitungsverfahren (Konvertierer) bereitgestellt, das Daten eines integrierten Modells, die ein dreidimensio-

naler Modellierer erzeugt hat, in Daten eines Modells umwandelt, die eine optimale Datenstruktur für die Zeichnungsbibliothek haben, und die in einem Aufzeichnungsmedium für Spiele gespeichert sind (oder simuliert werden), und die Zeichnungsbibliothek führt die Bildverarbeitung in Echtzeit aus. Daher kann man die Modelldaten im integrierten Formgebungsmodell-Format, das der dreidimensionale Modellierer verwendet, in einer Spieldatenmaschine einsetzen (oder in Simulatoren), und man kann eine natürlichere Bewegung von Objekten für Spiele bereitstellen.

[0172] Gemäß der Erfindung kann auch ein Bildverarbeitungsverfahren bereitgestellt werden, das wissentlich integrierte Formgebungsmodelle mit Hilfe von Konvertierungsdaten zeichnet, oder ein Aufzeichnungsmedium, das ein Programm für einen Computer aufzeichnet, damit das Bildverarbeitungsverfahren ausgeführt wird. Daher kann man Figuren, die aus integrierten Formgebungsmodellen bestehen, in Echtzeit zeichnen, und eine Spieldatenmaschine kann Bilder von Figuren mit einer natürlicheren Bewegung erzeugen.

Patentansprüche

1. Computerlesbares Aufzeichnungsmedium, das integrierte Formgebungs-Modelldaten aufzeichnet, geeignet zum Modellieren der Gliederbewegung eines Objekts, wobei zahlreiche Modelle (M1, ..., M5) über eine hierarchische Struktur verknüpft sind, und mindestens ein erstes Modell zahlreiche Ecken (V101, ..., V113, V501, ..., V506) aufweist, die Polygone bilden, und zumindest die Position einer ersten Ecke von den Positionen von zwei oder mehr der zahlreichen Modelle und von Gewichtungswerten aus diesen zwei oder mehr Modellen beeinflusst wird, und die integrierten Formgebungs-Modelldaten umfassen:

Formatdaten eines gemeinsamen Eckenpuffers, der Daten über die zahlreichen Ecken in den zahlreichen Modellen speichert, und zwar für jedes Modell in der hierarchischen Struktur;

eine Eckenliste (117), die für jedes der Modelle erzeugt wird, das die Ecken beeinflusst, und in der Eckendaten über eine Eckenkennung im gemeinsamen Eckenpuffer festgelegt sind;

Modellmatrixdaten (109), die für jedes der Modelle erzeugt werden und die Modellpositionen der zahlreichen Modelle in der Ursprungsform einstellen; und eine Polygonliste (116), die für jedes der Modelle erzeugt wird, das die Polygone aufweist, und die Polygondaten enthält, bei denen die Eckenkennung zugeordnete Daten sind,

wobei die Eckendaten in jeder Eckenliste (117) umfassen: mindestens die Positionsdaten einer Ecke, einen Gewichtungswert aus dem Modell, zu dem die Eckenliste gehört, und eine Eckenkennung im gemeinsamen Eckenpuffer, die der Ecke zugeordnet ist; dadurch gekennzeichnet, dass

Eckendaten der ersten Ecke, die von den Positionen der zwei oder mehr der zahlreichen Modelle beeinflusst werden, auf die Eckenlisten (117) der zwei oder mehr Modelle als verteilte Eckendaten aufgeteilt sind.

2. Computerlesbares Aufzeichnungsmedium nach Anspruch 1, das integrierte Formgebungs-Modelldaten aufzeichnet, wobei jede der Eckenlisten, die verteilte Eckendaten enthält, unterteilt ist in Anfangsgewichtsecken, für die eine Eckenoperation zuerst ausgeführt wird, wenn die hierarchische Struktur der Modelle durchlaufen wird, Mittelgewichtsecken, für die die Eckenoperation nicht zuerst oder zuletzt ausgeführt wird, und Endgewichtsecken, für die die Eckenoperation zuletzt ausgeführt wird.

3. Computerlesbares Aufzeichnungsmedium nach Anspruch 1 oder 2, das integrierte Formgebungs-Modelldaten aufzeichnet, wobei jede Eckenliste zudem Daten über eigene Ecken aufweist, die nur von dem Modell beeinflusst werden, zu dem diese Eckenliste gehört.

4. Computerlesbares Aufzeichnungsmedium nach irgendeinem der Ansprüche 1 bis 3, das integrierte Formgebungs-Modelldaten aufzeichnet, wobei die Eckendaten Koordinatendaten und Normalendaten der Ecken aufweist.

5. Computerlesbares Aufzeichnungsmedium nach Anspruch 4, das integrierte Formgebungs-Modelldaten aufzeichnet, wobei die Eckendaten einem lokalen Koordinatensystem des Modells der Eckenliste folgen, zu dem die Eckendaten gehören.

6. Computerlesbares Aufzeichnungsmedium nach irgendeinem der Ansprüche 1 bis 3, das integrierte Formgebungs-Modelldaten aufzeichnet, wobei die Eckenkennung im gemeinsamen Eckenpuffer eine Eintragsnummer der Ecke eines jeden Modells hat, sowie den Versatzwert, der der Anzahl der Ecken eines jeden Modells entspricht.

7. Computerlesbares Aufzeichnungsmedium nach irgendeinem der Ansprüche 1 bis 3, das integrierte Formgebungs-Modelldaten aufzeichnet, wobei jede der Polygonlisten einen Zeichnungsbefehl enthält, der das Zeichnen des Polygons des Modells anweist, oder einen Nicht-Zeichnungsbefehl, der angibt, dass das Polygon des Modells nicht zu zeichnen ist, zu dem diese Polygonliste gehört.

8. Bildverarbeitungsverfahren, das ursprüngliche Modelldaten in integrierte Formgebungs-Modelldaten umwandelt, geeignet zum Modellieren der Gliederbewegung eines Objekts, wobei zahlreiche Modelle (M1, ..., M5) über eine hierarchische Struktur verknüpft sind, und mindestens ein erstes Modell zahlreiche Ecken (V101, ..., V113, V501, ..., V506) auf-

weist, die Polygone bilden, und zumindest die Position einer ersten Ecke von den Positionen von zwei oder mehr der zahlreichen Modelle und von Gewichtungswerten aus diesen zwei oder mehr Modellen beeinflusst wird, und in den ursprünglichen Modelldaten zahlreiche Modelldaten über die hierarchische Struktur verknüpft sind, und jede der zahlreichen Modelldaten umfassen: eine Liste ursprünglicher Ecken (107), die die Eckendaten dieser Modelldaten aufweist, eine Gewichtungsliste (108), die Eckendaten über Ecken aufweist, die von diesen Modelldaten beeinflusst werden, und eine Liste ursprünglicher Polygone (106), in der die Eckenkennung der Eckenliste zugeordnete Daten sind, und

das Bildverarbeitungsverfahren die Schritte umfasst: das Erzeugen von Formatdaten eines gemeinsamen Eckenpuffers, damit Daten über die zahlreichen Ecken in den zahlreichen Modellen für jedes Modell in der hierarchischen Struktur gespeichert werden; das Erzeugen einer Eckenliste (117), die für jedes der Modelle erzeugt wird, das die Ecken beeinflusst, und in der Eckendaten über eine Eckenkennung im gemeinsamen Eckenpuffer festgelegt sind; und das Erzeugen einer Polygonliste (116), die für jedes der Modelle erzeugt wird, das die Polygone aufweist, bei denen die Eckenkennung zugeordnete Daten sind, wobei die Eckendaten in jeder Eckenliste (117) umfassen: mindestens die Positionsdaten der Ecken, Gewichtungswerte aus dem Modell, zu dem die Eckenliste gehört, und Eckenkennungen im gemeinsamen Eckenpuffer, die diesen Ecken zugeordnet sind; dadurch gekennzeichnet, dass der Schritt des Erzeugens einer Eckenliste für jedes Modell, das die Ecken beeinflusst, enthält:

- das Erzeugen von Eckendaten für die erste Ecke für jedes der zwei oder mehr der zahlreichen Modelle, das die erste Ecke beeinflusst; und
- das Verteilen der erzeugten Eckendaten auf die jeweiligen Eckenlisten (117) der zwei oder mehr Modelle als verteilte Eckendaten.

9. Bildverarbeitungsverfahren zum Umwandeln von Modelldaten nach Anspruch 8, wobei der Schritt des Erzeugens jeder der Eckenlisten umfasst, und zwar für jede Eckenliste, die verteilte Eckendaten enthält, das Unterteilen der Eckendaten in Daten für Anfangsgewichtsecken, für die eine Eckenoperation zuerst ausgeführt wird, wenn die hierarchische Struktur der Modelle durchlaufen wird, Daten für Mittelgewichtsecken, für die die Eckenoperation nicht zuerst oder zuletzt ausgeführt wird, und Daten für Endgewichtsecken, für die die Eckenoperation zuletzt ausgeführt wird, in der Eckenliste.

10. Bildverarbeitungsverfahren zum Umwandeln von Modelldaten nach Anspruch 8 oder 9, wobei die Eckenkennung eine Eintragsnummer der Ecke in jedem Modell hat, sowie einen Versatzwert, der der Anzahl der Ecken eines jeden Modells entspricht.

11. Bildverarbeitungsverfahren zum Umwandeln von Modelldaten nach irgendeinem der Ansprüche 8 bis 10, zudem umfassend einen Zeichnungsbefehl-Erzeugungsschritt, in dem: eine Dummy-Eckenoperation für die Eckendaten in der Eckenliste eines jeden Modells gemäß der Abarbeitungsfolge ausgeführt wird, die der hierarchischen Struktur der Modelle folgt;

wenn die Dummy-Eckenoperation für die Eckenliste eines Modells endet, ein Nicht-Zeichnungsbefehl in der Polygonliste dieses Modells erzeugt wird, falls die Gewichtungsberechnung für die Ecken dieses Modells noch nicht beendet ist; und ein Zeichnungsbefehl in der Polygonliste dieses Modells erzeugt wird, falls die Gewichtungsberechnung für die Ecken dieses Modells beendet ist.

12. Bildverarbeitungsverfahren zum Umwandeln von Modelldaten nach Anspruch 8, zudem umfassend einen Erzeugungsschritt für eine Liste einfacher gewichteter Ecken, wobei die erste Ecke mit den verteilten Eckendaten, die von den zwei oder mehr der zahlreichen Modelle beeinflusst wird, in eine einfache gewichtete Ecke umgewandelt wird, die nur von einem Modell beeinflusst wird, das von den zwei oder mehr Modellen den größten Einfluss ausübt, und die Daten bezüglich der umgewandelten einfach gewichteten Ecke auf die Eckenliste dieses Modells verteilt werden.

13. Bildverarbeitungsverfahren zum Zeichnen eines integrierten Formgebungsmodells, geeignet zum Modellieren der Gliederbewegung eines Objekts, wobei zahlreiche Modelle (M1, ..., M5) über eine hierarchische Struktur verknüpft sind, und mindestens ein erstes Modell zahlreiche Ecken (V101, ..., V113, V501, ..., V506) aufweist, die Polygone bilden, und zumindest die Position einer ersten Ecke von den Positionen von zwei oder mehr der zahlreichen Modelle und von Gewichtungswerten aus diesen zwei oder mehr Modellen beeinflusst wird, und die Daten des integrierten Formgebungsmodells umfassen:

Formatdaten eines gemeinsamen Eckenpuffers, der Daten über die zahlreichen Ecken in den zahlreichen Modellen speichert, und zwar für jedes Modell in der hierarchischen Struktur;

eine Eckenliste (117), die für jedes der Modelle erzeugt wird, das die Ecken beeinflusst, und in der Eckendaten über eine Eckenkennung im gemeinsamen Eckenpuffer festgelegt sind; und

eine Polygonliste (116), die für jedes der Modelle erzeugt wird, das die Polygone aufweist, und die Polygondaten enthält, bei denen die Eckenkennung zugeordnete Daten sind,

wobei die Eckendaten in jeder Eckenliste (117) umfassen: mindestens die Positionsdaten einer Ecke, einen Gewichtungswert aus dem Modell, zu dem die Eckenliste gehört, und die Eckenkennung im gemeinsamen Eckenpuffer, die der Ecke zugeordnet ist; und das Bildverarbeitungsverfahren die Schritte um-

fasst:

das Erzeugen des gemeinsamen Eckenpuffers entsprechend der zahlreichen Modelle in der Abarbeitungsfolge der hierarchischen Struktur gemäß den Formatdaten;

das Erzeugen von Modellmatrixdaten, in denen die Positionen der Modelle ausgehend von Spielfortschrittsdaten (53) eingestellt werden;

das Erzeugen von gemeinsamen Eckendaten durch das Ausführen einer Matrixberechnung zum Erzeugen von Eckendaten nach der Bewegung gemäß den Modellmatrixdaten (54), und einer Gewichtungsberechnung zum Integrieren von Gewichtungswerten aus den Modellen in die Eckendaten nach der Bewegung (55), und zwar für die Eckendaten der Eckenliste eines jeden Modells, und durch das Speichern oder Addieren dieser berechneten Eckendaten in Bereichen entsprechend den Eckenkennungen im gemeinsamen Eckenpuffer; und

das Rendern der Polygone gemäß den gemeinsamen Eckendaten; dadurch gekennzeichnet, dass Eckendaten der ersten Ecke, die von Positionen der zwei oder mehr der zahlreichen Modelle beeinflusst werden, auf die Eckenlisten (117) der zwei oder mehr Modelle als verteilte Eckendaten aufgeteilt werden.

14. Bildverarbeitungsverfahren nach Anspruch 13, worin im Erzeugungsschritt der gemeinsamen Eckendaten die Positionsdaten für die Matrixberechnung mit den Modellmatrixdaten multipliziert werden, und die multiplizierten Positionsdaten für die Gewichtungsberechnung mit dem Gewichtungswert multipliziert werden.

15. Bildverarbeitungsverfahren nach Anspruch 14, wobei jede der Eckenlisten unterteilt wird in: Anfangsgewichtsecken, für die eine Eckenoperation zuerst ausgeführt wird, wenn die Abarbeitung anhand der hierarchischen Struktur der Modelle erfolgt, Mittelgewichtsecken, für die die Eckenoperation nicht zuerst oder zuletzt ausgeführt wird, und Endgewichtsecken, für die die Eckenoperation zuletzt ausgeführt wird, und im Erzeugungsschritt der gemeinsamen Eckendaten die berechneten Eckendaten für die Anfangsgewichtsecken in Bereichen gespeichert werden, die den Eckenkennungen im gemeinsamen Eckenpuffer entsprechen, und die berechneten Eckendaten für die Mittelgewichtsecken und die Endgewichtsecken zu den Eckendaten in Bereichen addiert werden, die den Eckenkennungen im gemeinsamen Eckenpuffer entsprechen.

16. Bildverarbeitungsverfahren nach Anspruch 13 oder 14, wobei jede der Polygonlisten einen Zeichnungsbefehl aufweist, der das Zeichnen der Polygone des Modells anweist, zu dem die Polygonliste gehört, oder einen Nicht-Zeichnungsbefehl, der angibt, dass die Polygone des Modells, zu dem die Polygonliste gehört, nicht zu zeichnen sind, und der Renderschritt auf der Polygonliste des Modells be-

ruht, und zwar jedes Mal, wenn der Erzeugungsschritt der gemeinsamen Eckendaten eines jeden Modells endet, und das Rendern gemäß dem Zeichnungsbefehl ausgeführt wird.

17. Bildverarbeitungsverfahren nach Anspruch 16, zudem umfassend einen Freigabeschritt des Bereichs des gemeinsamen Eckenpuffers, der zu dem Modell gehört, für das das Rendern ausgeführt wird.

18. Bildverarbeitungsverfahren nach Anspruch 13, worin die Modelldaten auch eine Liste einfach gewichteter Ecken aufweisen, in der die erste Ecke, die von den zwei oder mehr der zahlreichen Modelle beeinflusst wird, in eine einfach gewichtete Ecke umgewandelt wird, die nur von dem Modell beeinflusst wird, das von den zwei oder mehr Modellen den größten Einfluss ausübt, und die Daten bezüglich der einfach gewichteten Ecke auf die Eckenliste dieses Modells verteilt werden, und das Bildverarbeitungsverfahren die gemeinsamen Eckendaten für ein vorbestimmtes Modell anhand der Liste einfach gewichteter Ecken erstellt.

19. Aufzeichnungsmedium, das ein Bildverarbeitungsprogramm zum Zeichnen eines integrierten Formgebungsmodells speichert, geeignet zum Modellieren der Gliederbewegung eines Objekts, wobei zahlreiche Modelle (M1, ..., M5) über eine hierarchische Struktur verknüpft sind, und mindestens ein erstes Modell zahlreiche Ecken (V101, ..., V113, V501, ..., V506) aufweist, die Polygone bilden, und zumindest die Position einer ersten Ecke von den Positionen von zwei oder mehr der zahlreichen Modelle und von Gewichtungswerten aus diesen zwei oder mehr Modellen beeinflusst wird,

und das Aufzeichnungsmedium Daten über das integrierte Formgebungsmodell aufzeichnet, und die Daten über das integrierte Formgebungsmodell umfassen:

Formatdaten eines gemeinsamen Eckenpuffers, der Daten über die zahlreichen Ecken in den zahlreichen Modellen speichert, und zwar für jedes Modell in der hierarchischen Struktur;

eine Eckenliste (117), die für jedes der Modelle erzeugt wird, das die Ecken beeinflusst, und in der Eckendaten über eine Eckenkennung im gemeinsamen Eckenpuffer festgelegt sind; und

eine Polygonliste (116), die für jedes der Modelle erzeugt wird, das solche Polygone aufweist, und die Polygondaten enthält, bei denen die Eckenkennung zugeordnete Daten sind,

wobei die Eckendaten in jeder Eckenliste (117) umfassen: mindestens die Positionsdaten einer Ecke, einen Gewichtungswert aus dem Modell, zu dem die Eckenliste gehört, und die Eckenkennung im gemeinsamen Eckenpuffer, die der Ecke zugeordnet ist; und worin das Bildverarbeitungsprogramm einen Programmcode enthält, der einen Computer veranlasst, folgende Prozeduren auszuführen:

das Erzeugen der gemeinsamen Eckenpuffer, die den zahlreichen Modellen zugeordnet sind, in der Abarbeitungsfolge der hierarchischen Struktur gemäß den Formatdaten;

das Erzeugen von Modellmatrixdaten, in denen die gegenseitigen Positionszusammenhänge zwischen den Modellen abhängig von den Spielfortschrittsdaten (**53**) eingestellt werden;

das Erzeugen von gemeinsamen Eckendaten durch das Ausführen einer Matrixberechnung zum Erzeugen von Eckendaten nach der Bewegung gemäß den Modellmatrixdaten (**54**), und einer Gewichtungsberechnung zum Integrieren der Gewichtungswerte aus den Modellen in die Eckendaten nach der Bewegung (**55**), und zwar für die Eckendaten der Eckenliste eines jeden Modells, und durch das Speichern oder Addieren dieser berechneten Eckendaten in Bereichen entsprechend den Eckenkennungen im gemeinsamen Eckenpuffer; und

das Rendern der Polygone gemäß den gemeinsamen Eckendaten; dadurch gekennzeichnet, dass Eckendaten der ersten Ecke, die von Positionen der zwei oder mehr der zahlreichen Modelle beeinflusst werden, auf die Eckenlisten (**117**) der zwei oder mehr Modell als verteilte Eckendaten aufgeteilt werden.

20. Aufzeichnungsmedium nach Anspruch 19, das ein Bildverarbeitungsprogramm speichert, wobei in der Erzeugungsprozedur der gemeinsamen Eckendaten die Positionsdaten für die Matrixberechnung mit den Modellmatrixdaten multipliziert werden, und die multiplizierten Positionsdaten für die Gewichtungsberechnung mit dem Gewichtungswert multipliziert werden.

21. Aufzeichnungsmedium nach Anspruch 20, das ein Bildverarbeitungsprogramm speichert, wobei jede der Eckenlisten unterteilt ist in Anfangsgewichtsecken, für die eine Eckenoperation zuerst ausgeführt wird, wenn die hierarchische Struktur der Modelle durchlaufen wird, Mittelgewichtsecken, für die die Eckenoperation nicht zuerst oder zuletzt ausgeführt wird, und Endgewichtsecken, für die die Eckenoperation zuletzt ausgeführt wird, und in der Erzeugungsprozedur der gemeinsamen Eckendaten die berechneten Eckendaten für die Anfangsgewichtsecken in Bereichen gespeichert werden, die den Eckenkennungen im gemeinsamen Eckenpuffer entsprechen, und die berechneten Eckendaten für die Mittelgewichtsecken und die Endgewichtsecken zu den Eckendaten in den Bereichen addiert werden, die den Eckenkennungen im gemeinsamen Eckenpuffer entsprechen.

22. Aufzeichnungsmedium nach Anspruch 19 oder 20, das ein Bildverarbeitungsprogramm speichert, wobei jede der Polygonlisten einen Zeichnungsbefehl aufweist, der das Zeichnen der Polygone des Modells anweist, zu dem die Polygonliste gehört, oder einen Nicht-Zeichnungsbefehl, der angibt,

dass die Polygone des Modells, zu dem die Polygonliste gehört, nicht zu zeichnen sind, und die Renderprozedur auf der Polygonliste des Modells beruht, und zwar jedes Mal, wenn die Erzeugungsprozedur der gemeinsamen Eckendaten eines jeden Modells endet, und das Rendern gemäß dem Zeichnungsbefehl ausgeführt wird.

23. Aufzeichnungsmedium nach Anspruch 22, das das Bildverarbeitungsprogramm speichert, zudem umfassend eine Freigabeprozedur für den Bereich des gemeinsamen Eckenpuffers, der zu dem Modell gehört, für das das Rendern ausgeführt wird.

24. Aufzeichnungsmedium nach Anspruch 19, das das Bildverarbeitungsprogramm speichert, worin die Modelldaten auch eine Liste einfach gewichteter Ecken aufweisen, in der die erste Ecke, die von den zwei oder mehr der zahlreichen Modelle beeinflusst wird, in eine einfach gewichtete Ecke umgewandelt wird, die nur von dem Modell beeinflusst wird, das von den zwei oder mehr Modellen den größten Einfluss ausübt, und Daten der in eine einfach gewichtete Ecke umgewandelten Ecke in die Eckenliste dieses Modells verteilt werden, und das Bildverarbeitungsprogramm einen Code enthält, der den Computer veranlasst, eine Erzeugungsprozedur für die gemeinsamen Eckendaten auszuführen, und zwar für ein vorbestimmtes Modell gemäß der Liste einfach gewichteter Ecken.

Es folgen 24 Blatt Zeichnungen

Integriertes Formgebungsmodell (Hüllen)

EP00301217.6

FIG. 1A Grundform

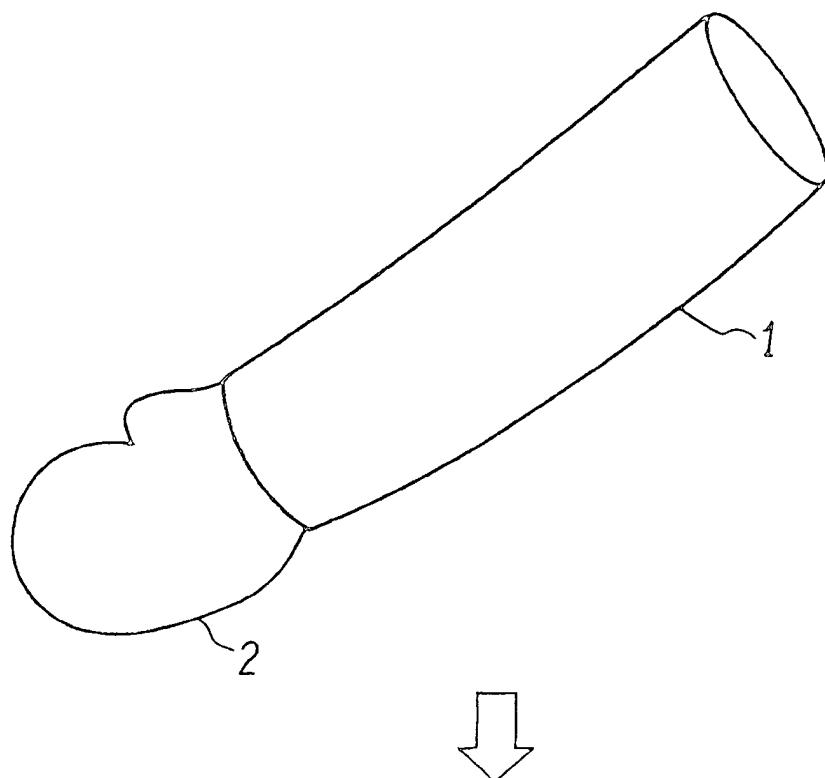


FIG. 1B Bewegte Form

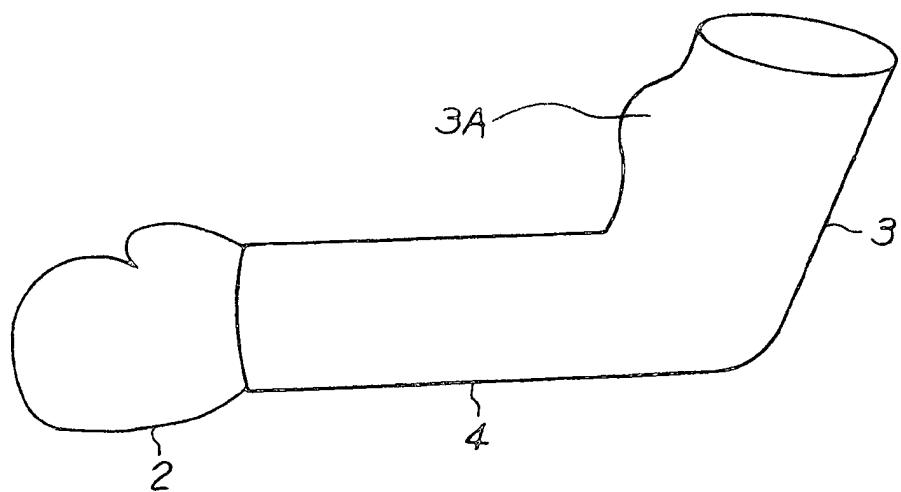
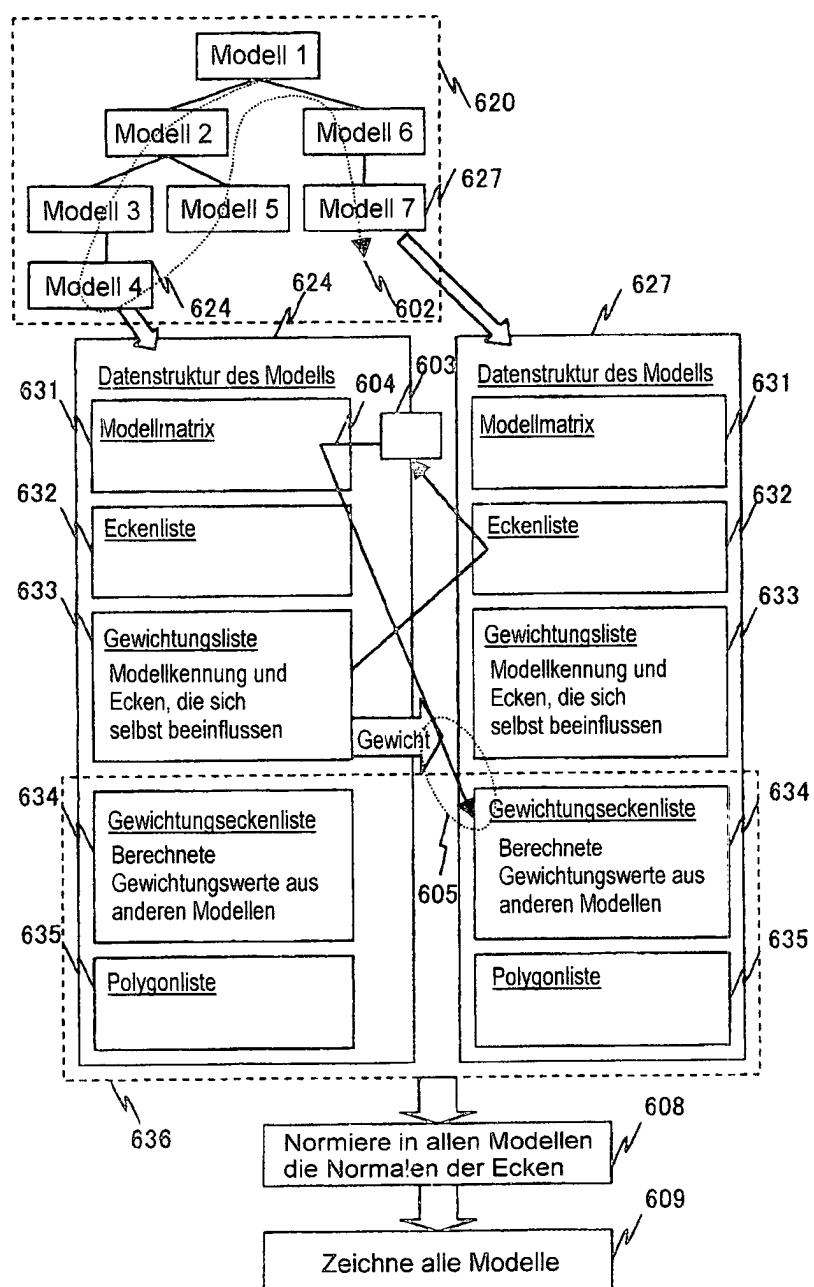


FIG. 2



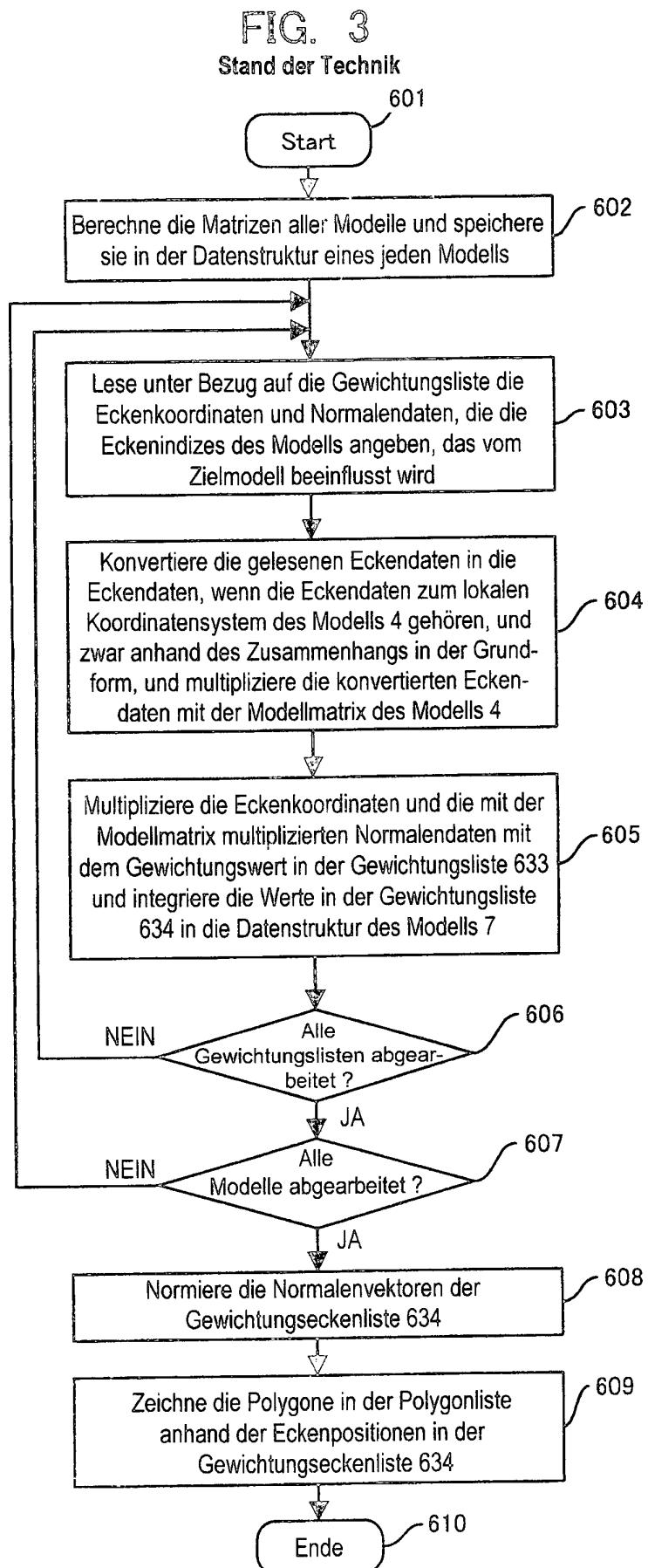


FIG. 4
Integriertes Formgebungsmodell (Hüllen)

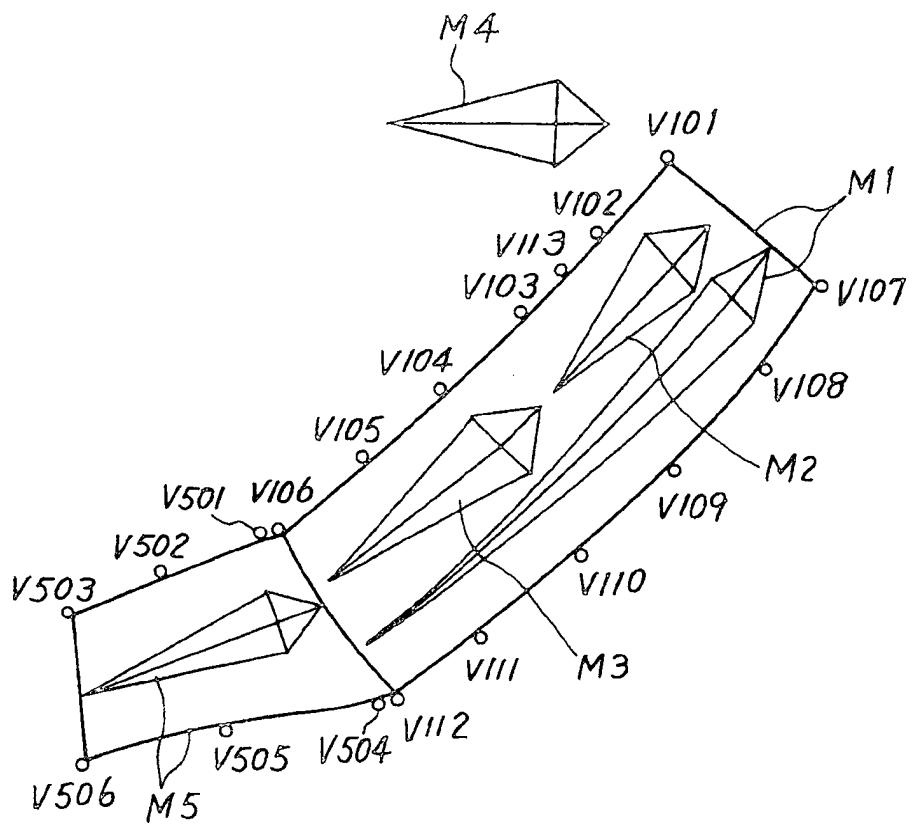


FIG. 5

Datenanordnung des integrierten Formgebungsmodells

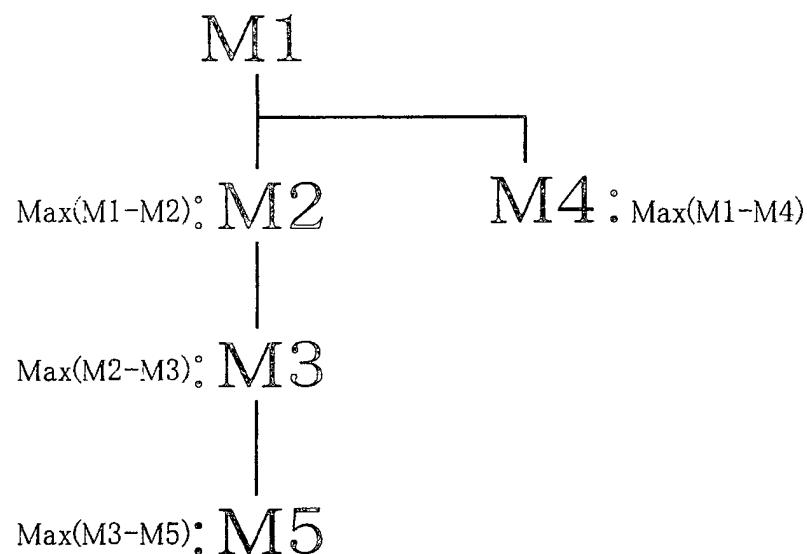
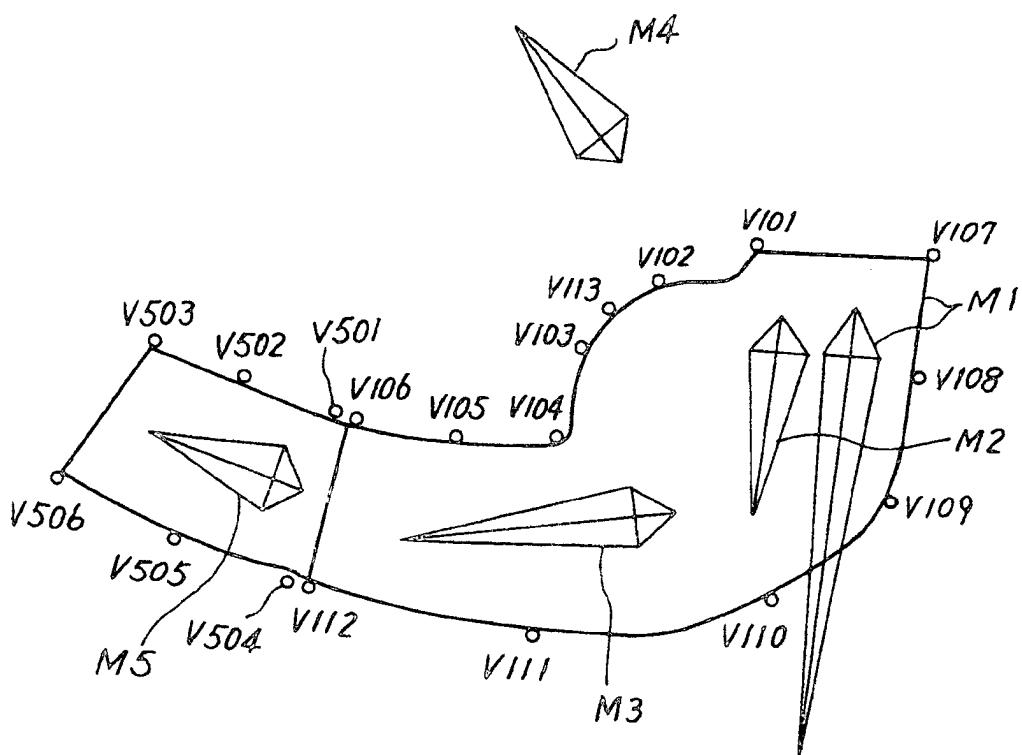


FIG. 6

Modell	Beeinfluss. Modell	M1	M2	M3	M4	M5
	Ecke					
M1	V101	100%				
	V102		30%	30%	40%	
	V103		30%	30%	40%	
	V104		40%	60%		
	V105		40%	60%		
	V106			100%		
	V107	100%				
	V108		60%	40%		
	V109		60%	40%		
	V110		40%	60%		
	V111		40%	60%		
	V112			100%		
	V113				100%	
M5	V501					100%
	V502					100%
	V503					100%
	V504					100%
	V505					100%
	V506					100%

FIG. 7

Bewegung des integrierten Formgebungsmodells (Hülle)



Gewichtungsberechnung

FIG. 8A Grundform

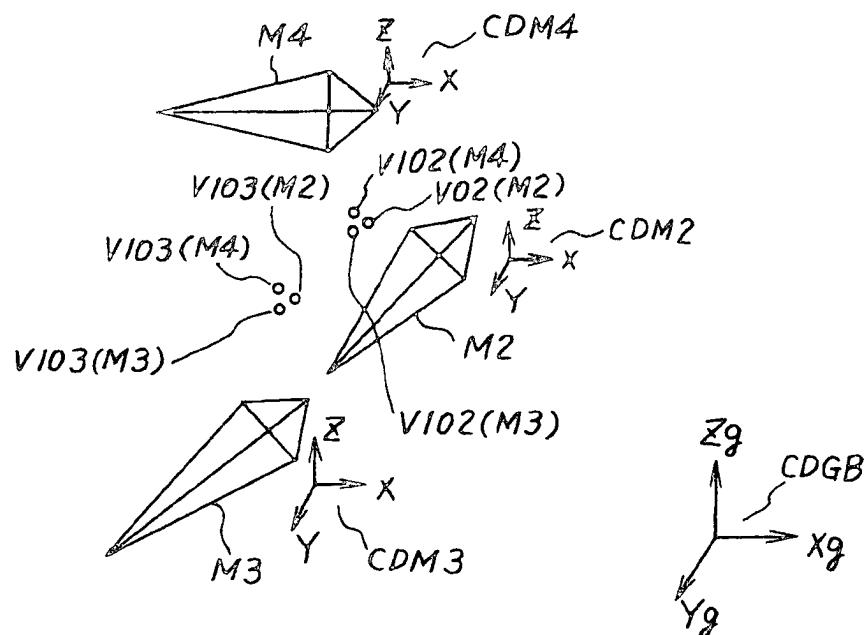


FIG. 8B Bewegte Form

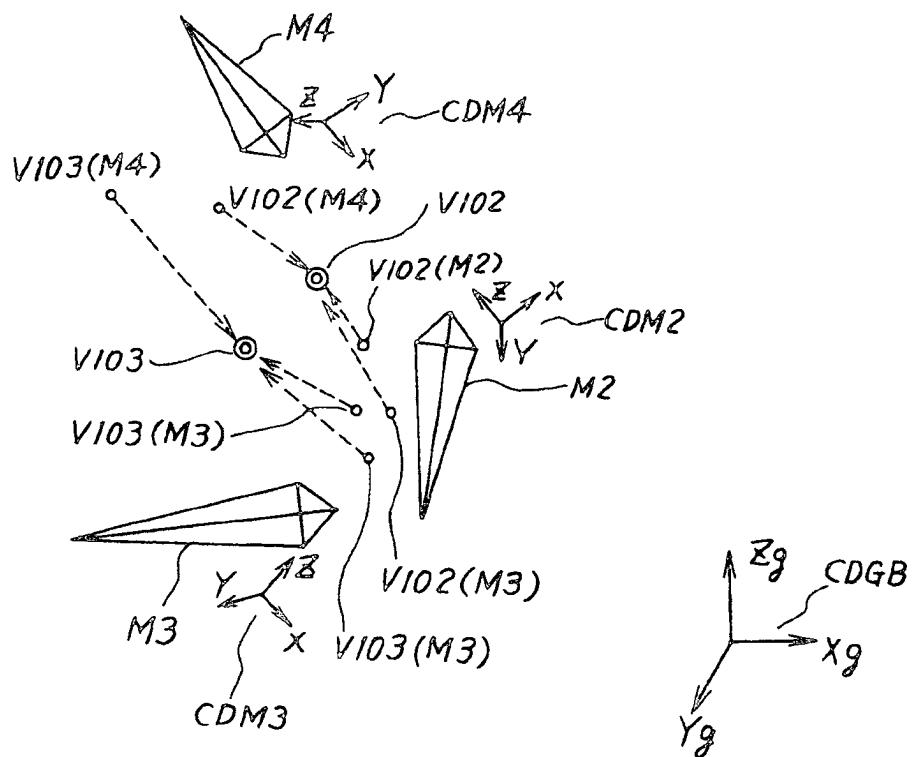


FIG. 9

Berechnung der Modellbewegung (Matrizenberechnung) und Gewichtungsberechnung

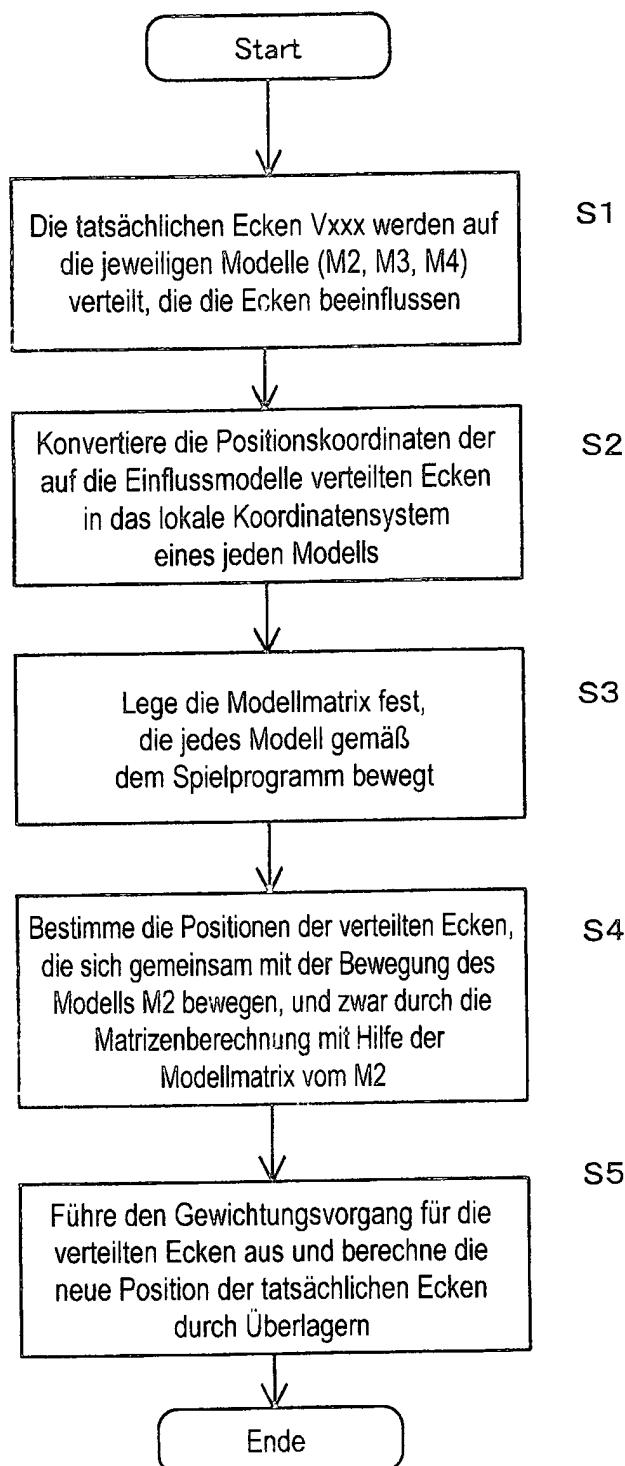
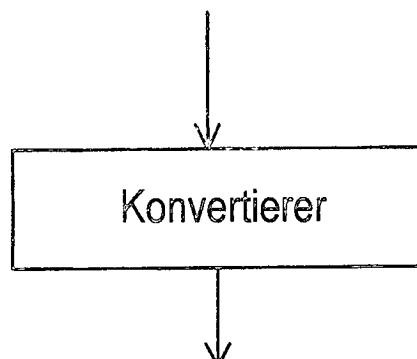
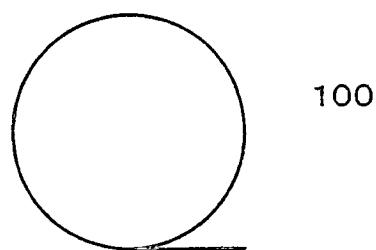
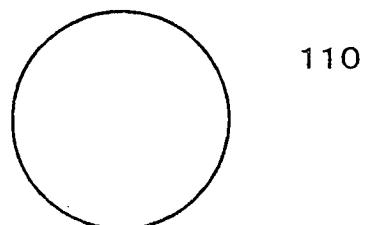


FIG. 10

Ursprungsdaten des integrierten Formgebungsmodells gemäß
dem 3D-Modellierer



Konvertierungsdaten für die Zeichnungsbibliothek



Aufzeichnung auf CDROM (Aufzeichnungsmedium für das Spiel)

FIG. 11

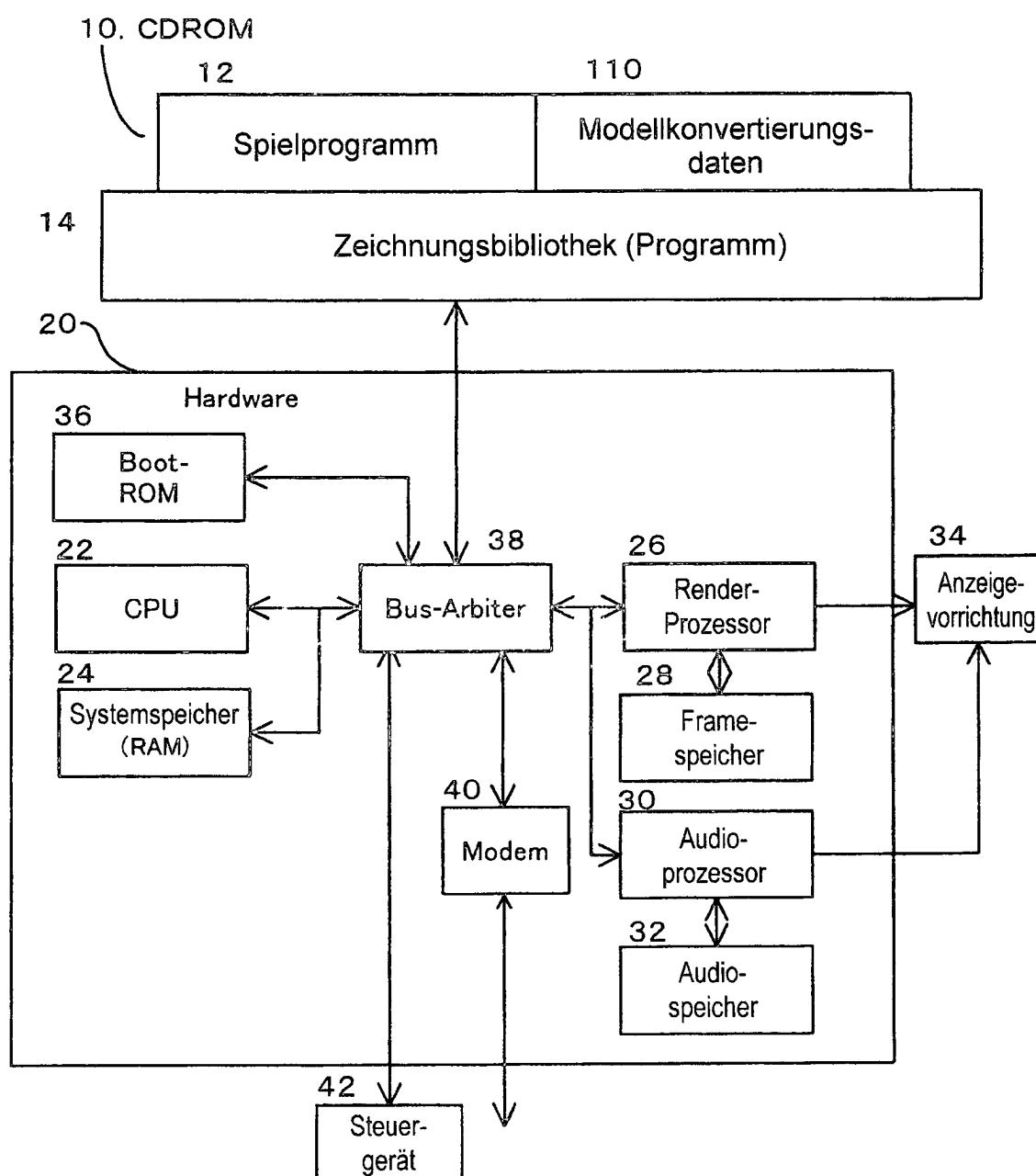


FIG. 12
Konvertierer und Konvertierungsdaten

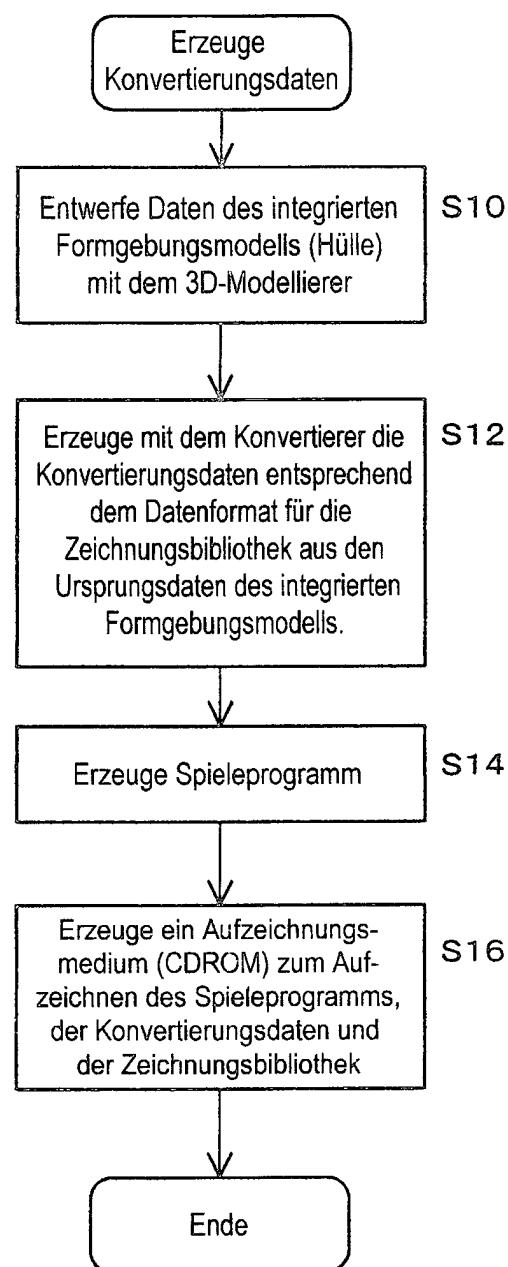


FIG. 13
Spielemaschine und Zeichnungsbibliothek

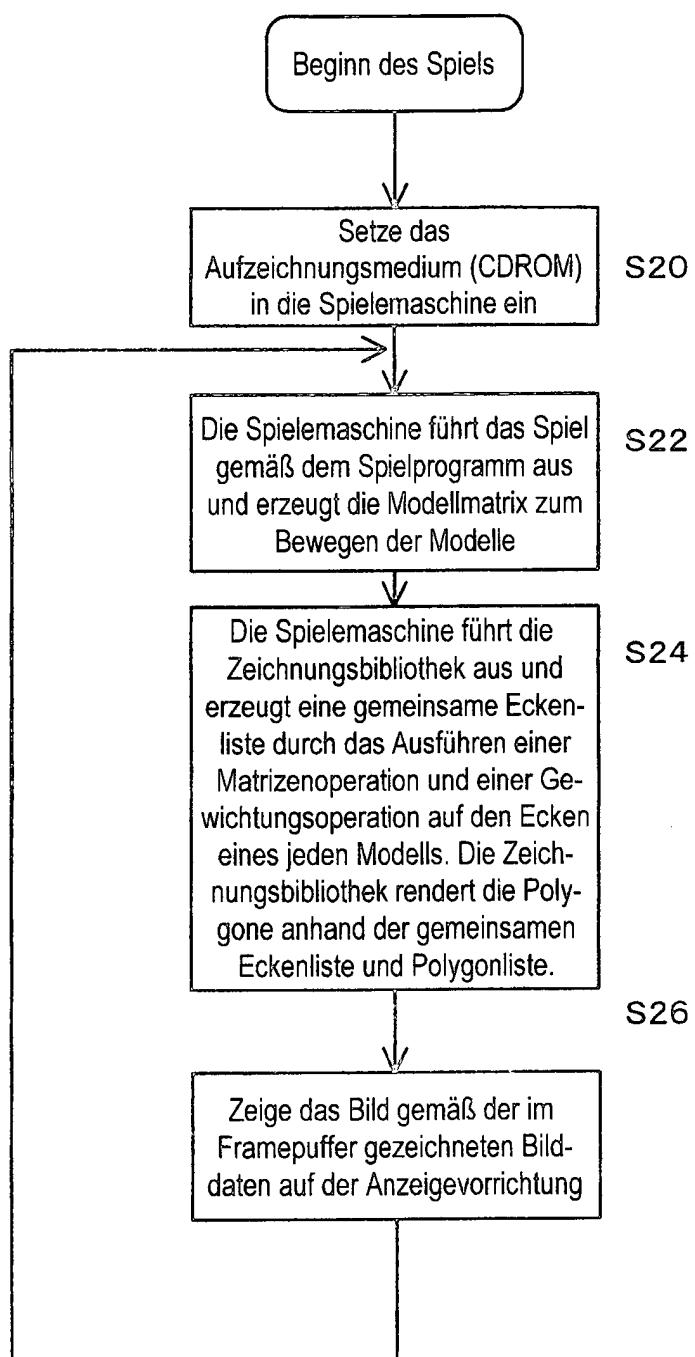


FIG. 14
Prozedur des Konvertierers (1)

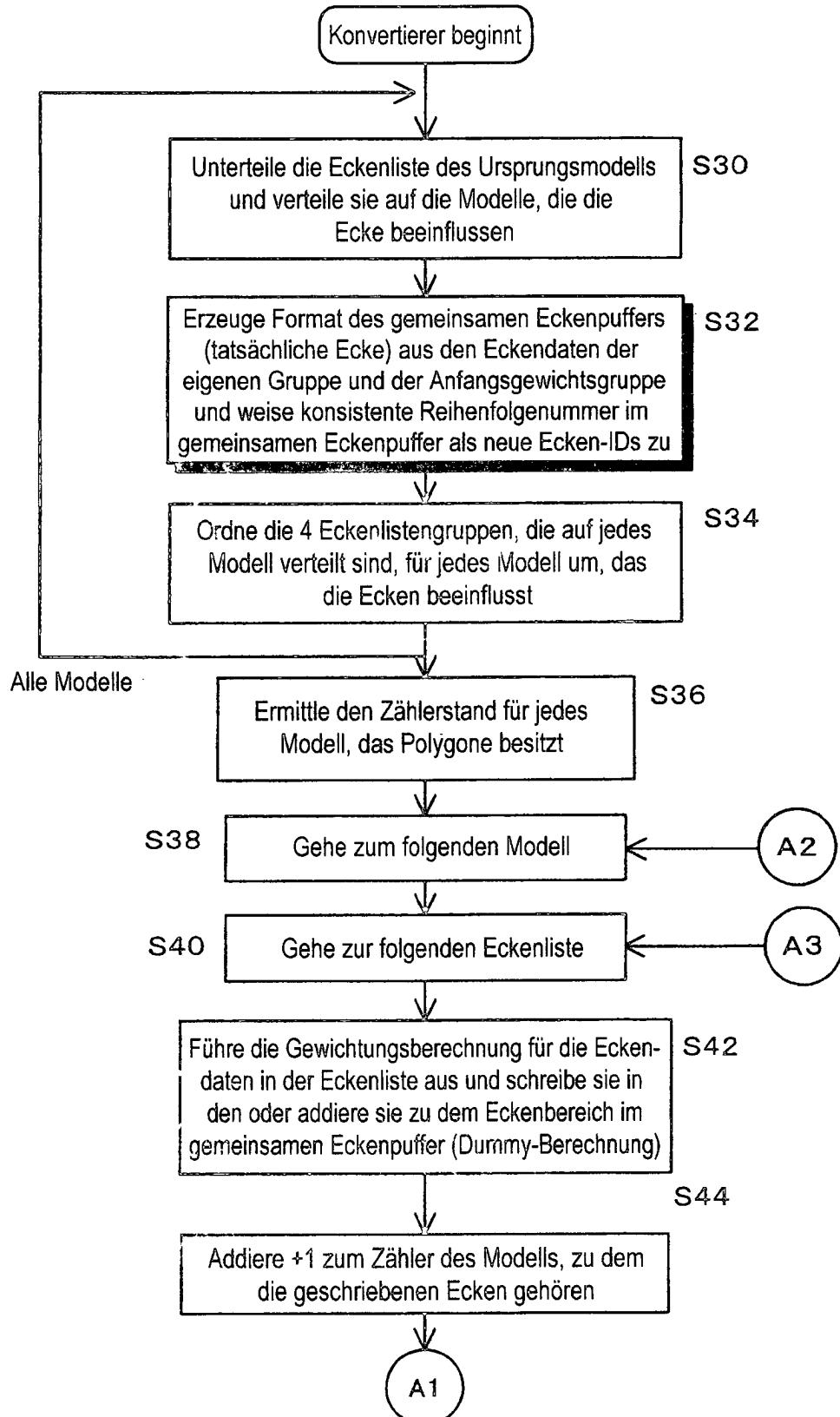


FIG. 15
Prozedur des Konvertierers (2)

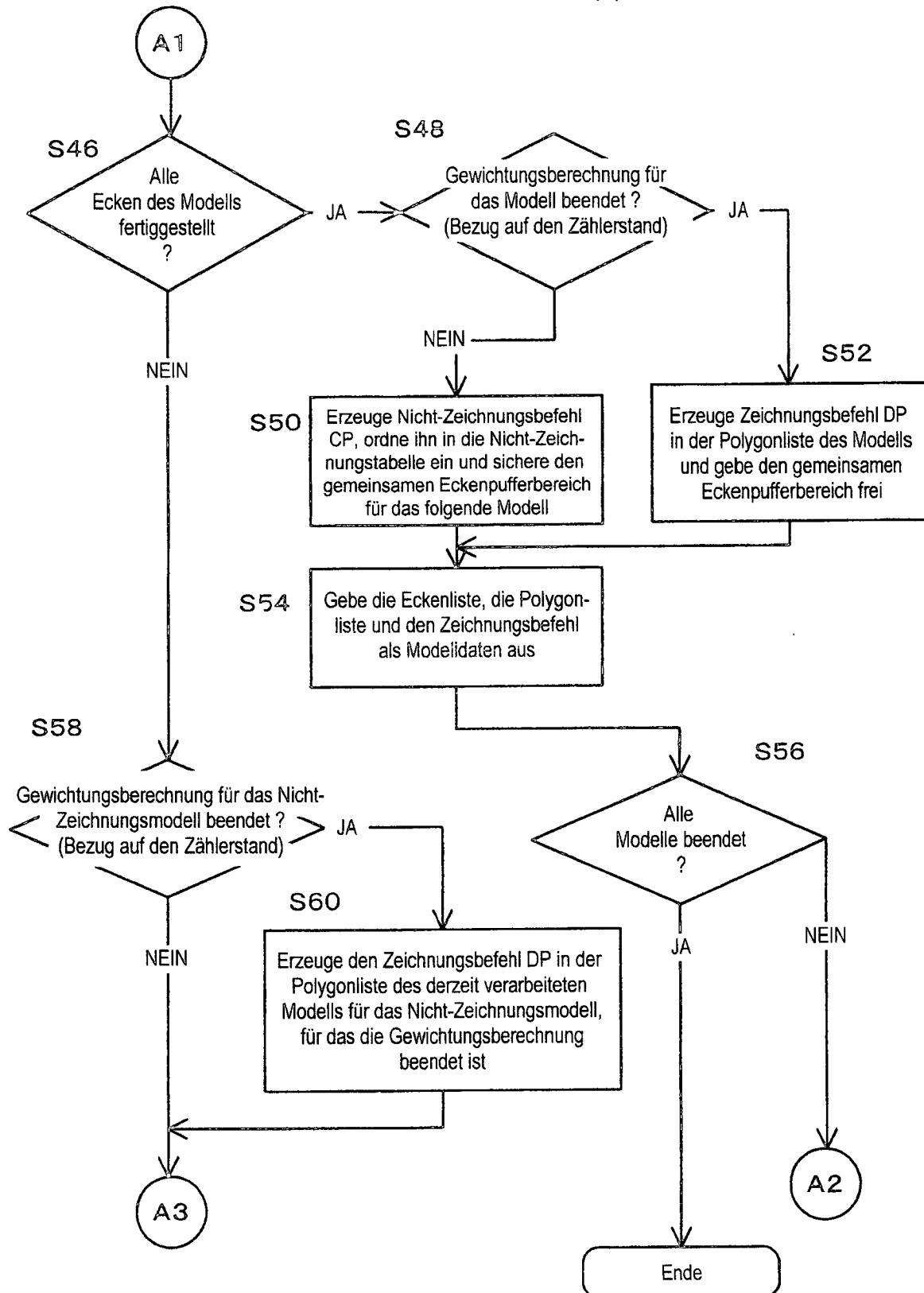


FIG. 16
 Verteilung der tatsächlichen Ecken auf die Modelle
 und Erzeugung der Eckenliste

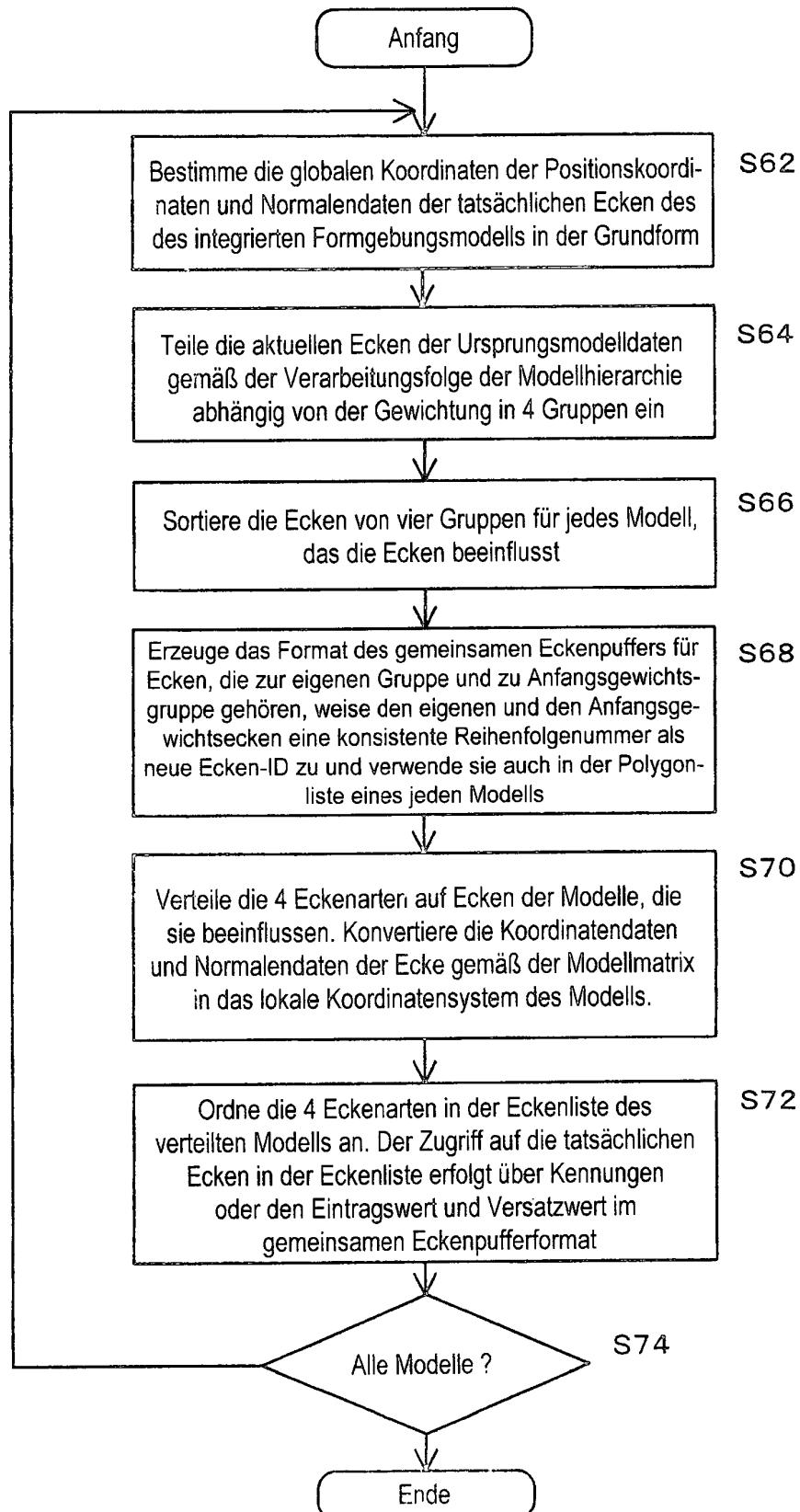


FIG. 17
Konvertiere die Ursprungsdaten in Konvertierungsdaten

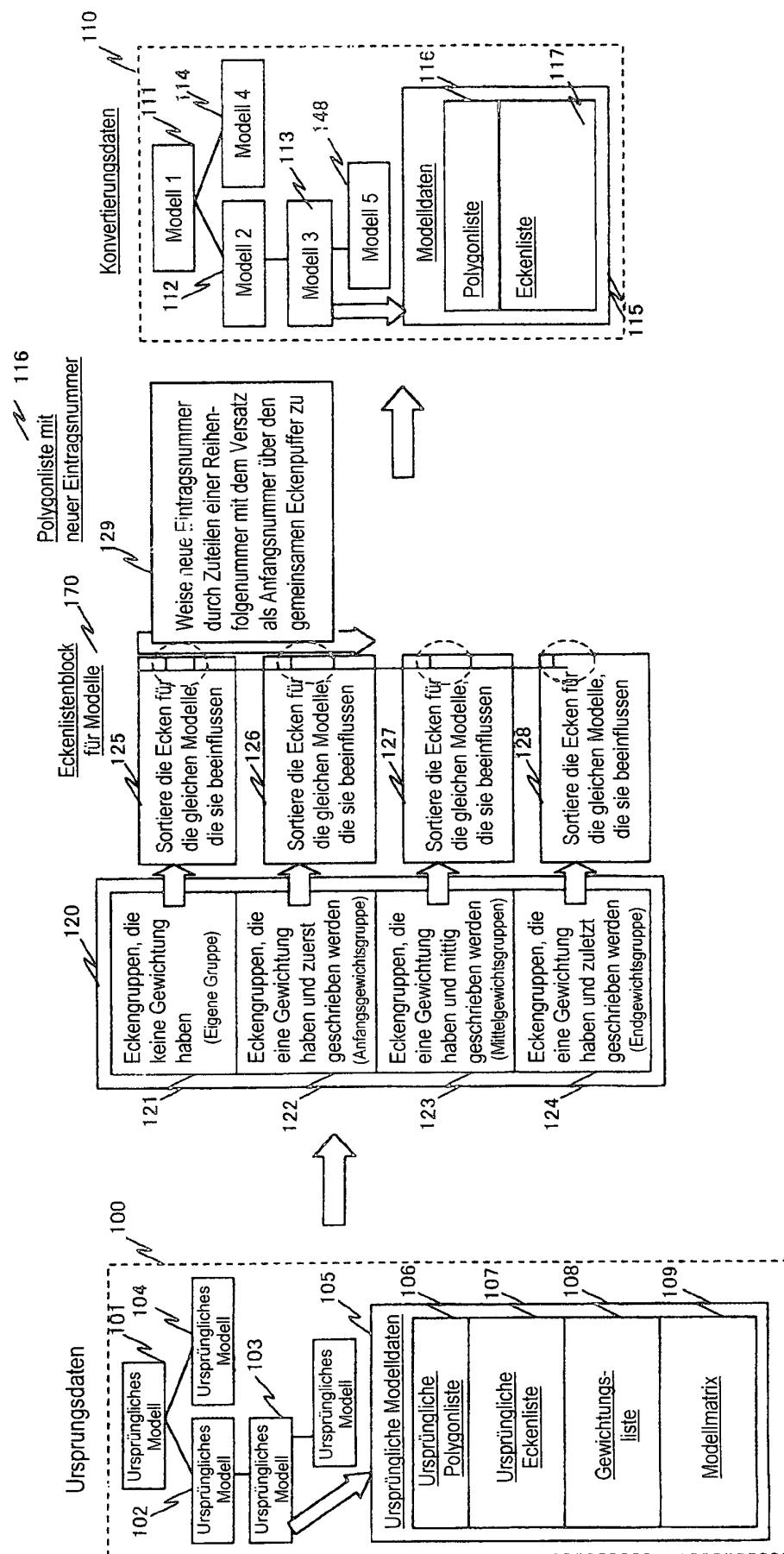


FIG. 18

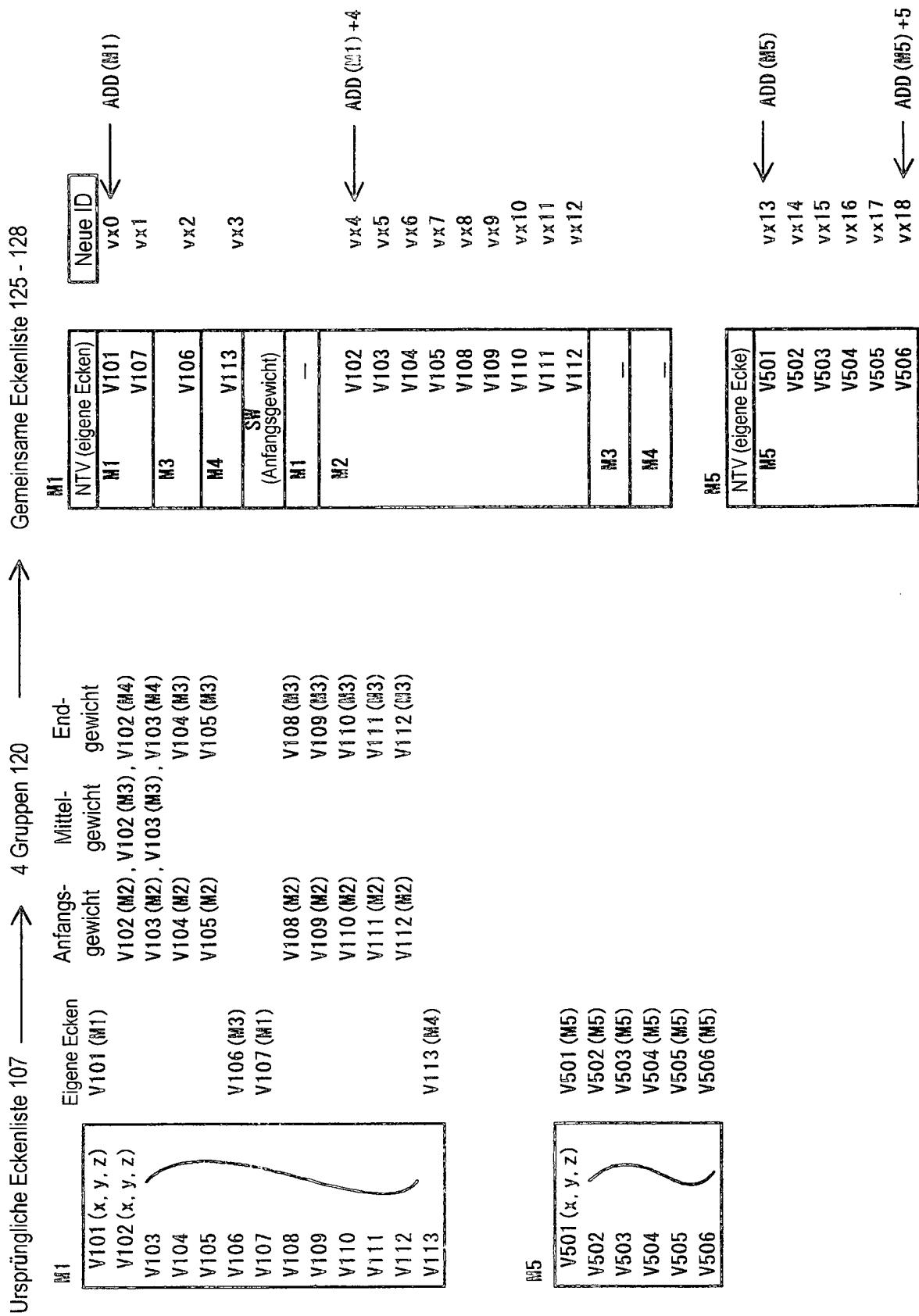


FIG. 19
Eckenliste 117

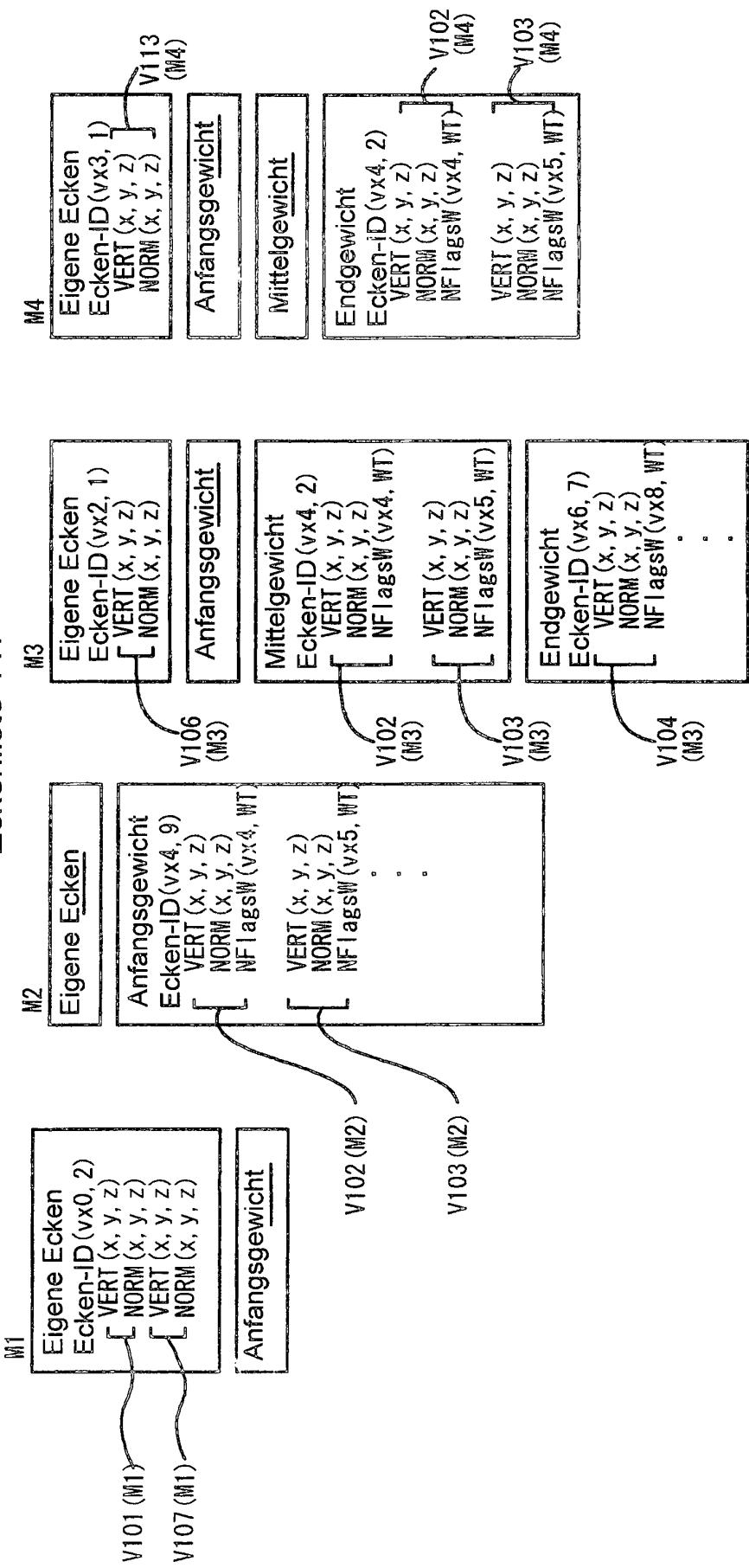
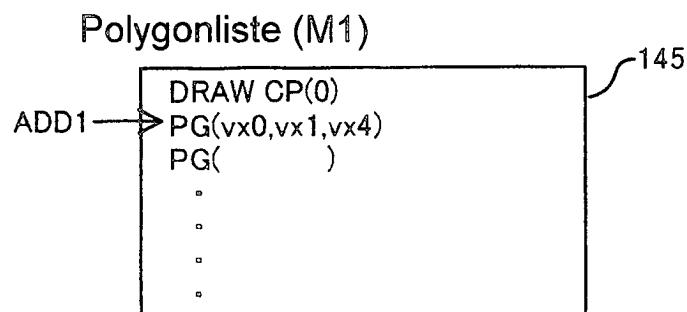
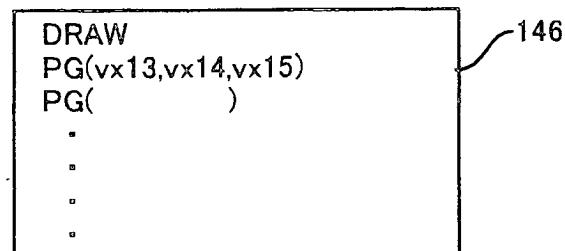


FIG. 20



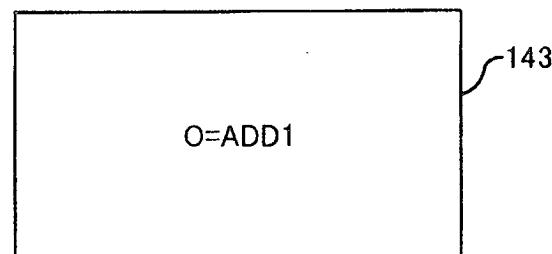
Polygonliste (M5)



Polygonliste (M4)



FIG. 21 Nicht-Zeichnungstabelle



Ecken-Kennung

FIG. 22A
Gemeinsamer Eckennpuffer

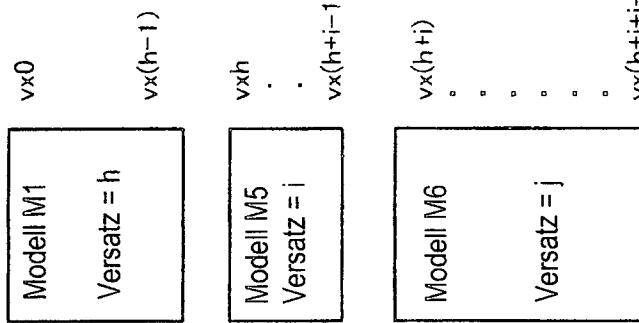


FIG. 22B
Erzeugung und Freigabe des gemeinsamen Eckennpuffers

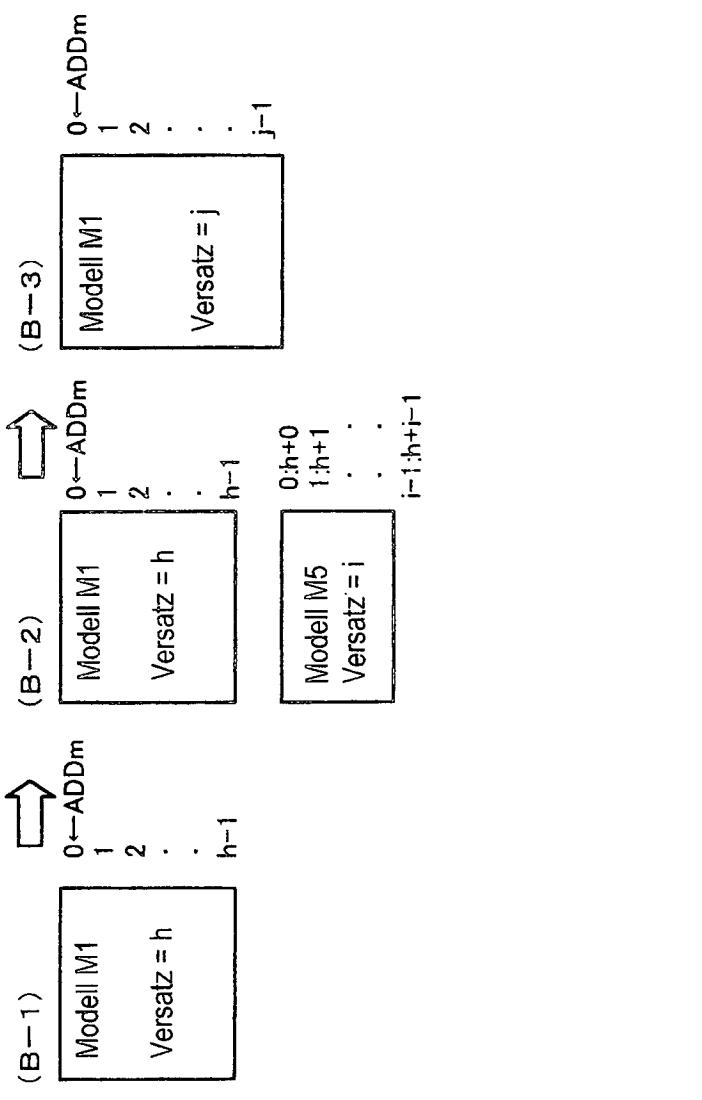


FIG. 23
Beispiel für eine Eckenliste

```

200
VLIST 201 vertex_list1[]
START 211 212 213 218
Cnkv_VN(FV_CNT, 55),
214 215
216 OffnbIdx(98, 9),
217 VERT( 0x408fc1d7, 0x3e94c1ef, 0xbff01b247 ),
NORM( 0x3f01fa3b, 0x3f327e43, 0xbff018ba8 ),
VERT( 0x408fcdb, 0x3ec104ab, 0x3c910610 ),
NORM( 0x3ee6e734, 0x3f641d71, 0x3d50e92e ),
VERT( 0x409a3fed, 0xbc9d3460, 0xbb9f18c0 ),
NORM( 0x3f7f633c, 0xbd8d5e8c, 0xbb730d7c ),
221 212 222 213
Cnkv_VN_NF(FV_CNT|FW_START, 169),
214 215 228
216 OffnbIdx(108, 24), 226 227
217 VERT( 0x404d6e16, 0x3effb6d1, 0xbff8b71bd ),
NORM( 0x3d188593, 0x3f44cfcc, 0xbff236ecc ),
NFlagsW{ 0, | 49. 983261 },
VERT( 0x40879d8c, 0x3f02c228, 0x3f81db3f ),
NORM( 0x3dcdf0e8, 0x3f49da46, 0x3f1b5a48 ),
NFagsW{ 1, | 22. 523094 },
VERT( 0x4087948e, 0xbba94240, 0x3fa30fa6 ),
NORM( 0x3dfa8d19, 0xbcd3a631e, 0x3f7dcf4e ),
NFagsW{ 3, | 26. 207253 },
221 212 222 213
Cnkv_VN_NF(FV_CNT|FW_MIDDLE, 169),
214 215 238
216 OffnbIdx(132, 24), 226 227
217 VERT( 0x404d92df, 0xbef36b61, 0x3f866400 ),
NORM( 0x3d034aec, 0xbef5ee707, 0x3efb4258 ),
NFagsW{ 0, | 49. 985958 },
VERT( 0x404d9689, 0xbef4533b, 0xbff8db2cd ),
NORM( 0x3d16b159, 0xbff4f1bbf, 0xbff162d8f ),
NFagsW{ 1, | 49. 985447 },
VERT( 0x4000e897, 0xbff148850, 0xbdd1a7260 ),
NORM( 0x3d94da54, 0xbff7ee86c, 0xbdd6ae4c6 ),
NFagsW{ 2, | 93. 426033 },
VERT( 0x404d9c83, 0xbff150deb, 0xbdd1c5780 ),
NORM( 0x3cb3ddd3, 0xbff7f8586, 0xbdd699996 ),
NFagsW{ 3, | 49. 948112 },
221 212 242 213
Cnkv_VN_NF(FV_CNT|FW_END, 169),
214 215 248
216 OffnbIdx(0, 24), 226 227
217 VERT( 0x3c9e3800, 0xbef388ea, 0x3f866400 ),
NORM( 0x3cbc8797, 0xbef3bef998, 0x3efb4258 ),
NFagsW{ 132, | 50. 014042 },
VERT( 0x3ca26200, 0xbef4703a, 0xbff8db2cd ),
NORM( 0x3ce98fb8, 0xbff4f31a3, 0xbff162d8f ),
NFagsW{ 133, | 50. 014545 },
VERT( 0xbff793a48, 0xbff11fd7a, 0xbdd1a7260 ),
NORM( 0x3d7e56dc, 0xbff7f1579, 0xbdd6ae4c6 ),
NFagsW{ 134, | 6. 573956 },
202
CnkEnd()
END 203

```

FIG. 24

Prozedur der Zeichnungsbibliothek

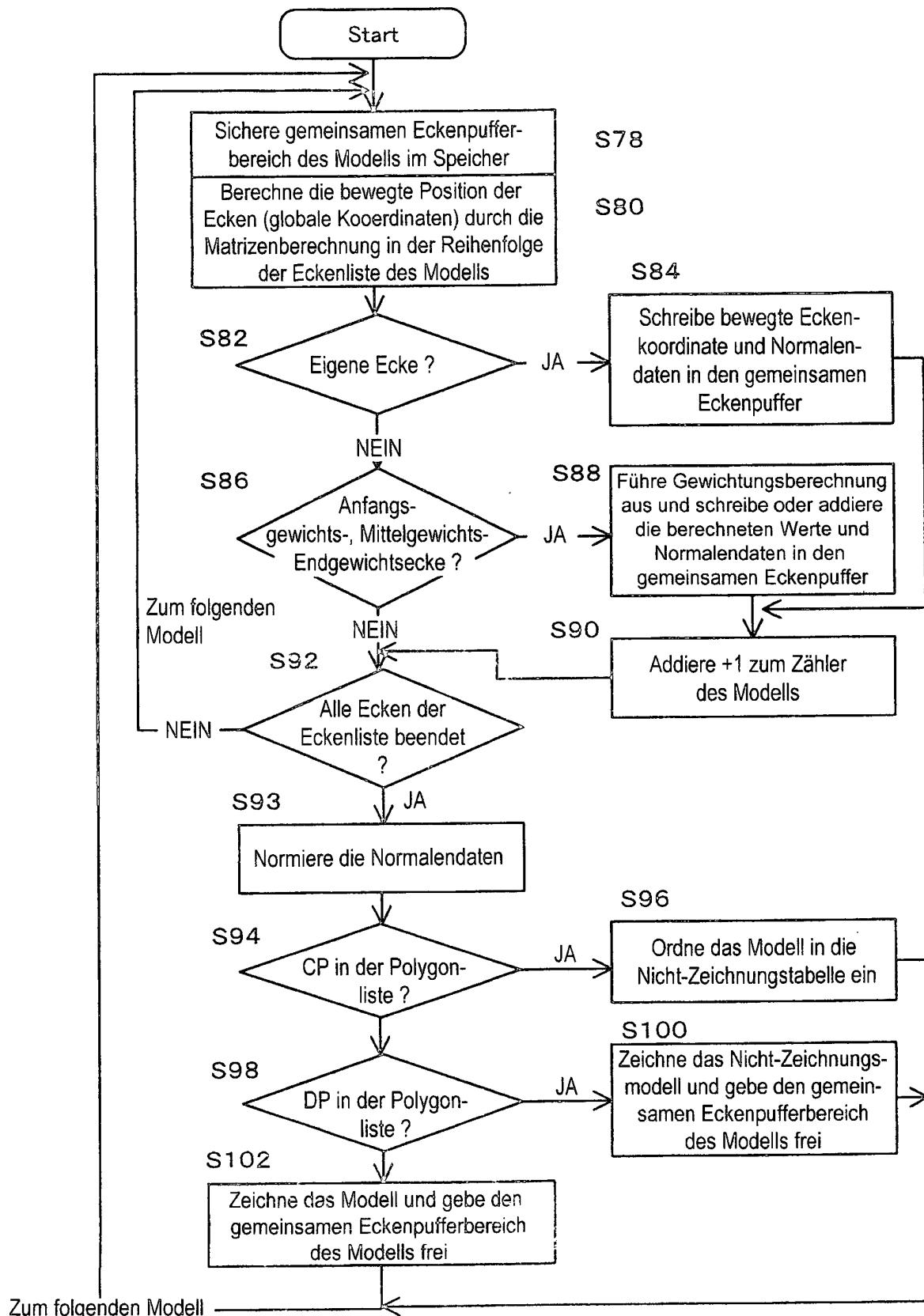


FIG. 25
Einfach gewichtete Ecke

