

Theorie zu DDD.TTP

Vorwort

*"If it doesn't fit - use a bigger hammer."
Chief Engineer Miles O'Brien*

Vor einigen Monaten sah ich das erste Mal eines dieser 3-dimensionalen Bilder. Eins von denen, wo man nix sieht und nach ein paar Tagen ist einem richtig schlecht und dann geht's doch. Das heißt, eigentlich sah ich eine Freundin, die in einen Copy-Shop lief und sagte, sie müsse etwas völlig Geniales kopieren, bevor sie es zurückgeben muß. Ich war neugierig und ging mit. Das völlig geniale war ein Poster, auf dem ich nur Unsinn sah. Sie sagte, es wäre ein 3-dimensionales Bild. Ich bin ein recht naiver Mensch und so glaubte ich ihr das und machte mir auch eine Kopie. Später saß ich zu Hause, starrte auf die Wand, an der das Poster hing und fühlte mich etwas veralbert. Aber nur für ein paar Stunden. Dann war's um mich geschehen. Ich sah es, ich war begeistert und ich wollte mehr! Zu der Zeit (es war im Februar '94) gab es in Deutschland noch nichts derartiges, es wurde von jemand in Köln berichtet, der sowas bei einer Amerika-Reise besorgt hatte, und die ersten Poster tauchten in Amsterdam bei Straßen-Händlern auf.

Nun ist es Mai 1994 und in einem Monat kommt das zweite Buch auf den deutschen Markt und die Poster sind auch bei uns zu kaufen. Mein Interesse an diesen Bildern nimmt immer noch zu. Allerdings hat es sich etwas verlagert. Während ich - wie wohl jeder, der sie kennengelernt hat - zu Anfang jedes Bild einfach nur sehen wollte, rückte langsam die Frage nach dem „Wie“ in den Vordergrund.

Das ist der Grund, warum ich das Programm DDD geschrieben habe. Ich wollte Antworten haben auf die Fragen: Wie kann in einem 'ganz normalen' Poster ein plastisches Bild enthalten sein? Und wie ist dies zu programmieren?

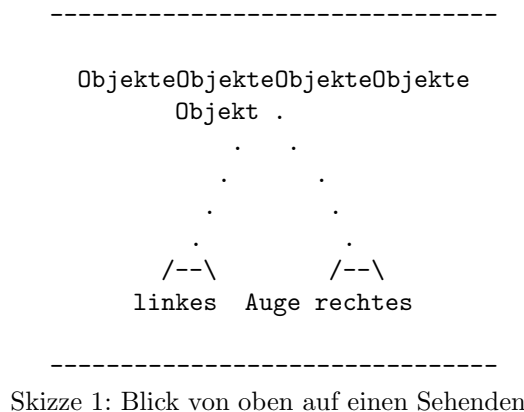
Während das Programm die Anschauung liefert und Spielereien mit Grafiken ermöglicht, gibt dieser Text die Antwort auf die genannten Fragen, so, wie ich sie mir beim Schreiben des Programms gefunden habe. Es ist durchaus möglich, daß es verschiedene Wege gibt, das Prinzip zu erklären. Daher wäre es interessant, andere Meinungen und Begründungen kennenzulernen, Vorschläge zur Verbesserung des Programms zu hören und Ideen wie Umsetzungen für 3-dimensionale Bilder zu sammeln.

Wolfgang Westerboer
Tannenbergstr. 23
48147 Münster

westerb@math.uni-muenster.de

1 3–dimensionales Sehen

Jeder Mensch hat in seinem Kopf eine 3-dimensionale Vorstellung der Welt. Wie kommt er dazu, wo doch im Auge nur ein 2-dimensionales Bild auf der Netzhaut entsteht? Wenn wir ein Auge schließen, ist die Welt auf einmal viel weniger interessant. Nun, der Trick ist einfach: Wir haben (meistens) 2 Augen. Diese liefern uns gleichzeitig Informationen über den Teil der Welt, den wir gerade sehen. Diese Informationen beschreiben fast denselben Ausschnitt. Aber eben nur fast. Obwohl die Augen nur um ein paar cm auseinanderstehen, nehmen sie doch verschiedene Bilder auf. Hier zur Veranschaulichung eine Skizze:



Das Gehirn – ein feines Ding, auf das viel weniger Menschen verzichten sollten, als dieses der Fall ist – dieses Gehirn also verknüpft und interpretiert die unterschiedlichen Informationen so, daß es daraus eine plastische Vorstellung der Welt konstruiert. Und das ist es, was wir bewußt wahrnehmen.

Aus der Fülle der Informationen werden sehr verschiedene genutzt, um eine Vorstellung zu gewinnen, wie die Welt nun wirklich aussieht:

- Die verschiedenen visuellen Wahrnehmungen (Skizze 1: das linke Auge sieht das 'e' nicht, das das rechte Auge sieht, weil etwas davor steht und es verdeckt.)
- Auch der Winkel, in dem die Augen zueinander stehen, ist eine Informationsquelle (kann man sich ebenfalls mit Hilfe von Skizze 1 vorstellen).
- Aus Krümmung der Linse (wo fokussieren wir), erhalten wir ebenfalls Informationen.
- Schlagt mich nicht: mehr fällt mir nicht ein. Ich bin halt kein Gehirn und weiß nicht, wie das Ding das macht. Das wichtigste für unsere Sache haben wir aber beisammen.

Es sollte klar geworden sein, daß das Gehirn nur eine Vorstellung schafft, die die Informationen, die es von den Augen erhält, sinnvoll deutet. Es ist für uns wesentlich, daß diese Vorstellung auch völliger Unsinn sein kann. Nämlich dann, wenn wir ein Bild betrachten, daß ja nun mal 2–dimensional ist, und trotzdem etwas 3–dimensionales sehen.

Es gibt verschiedene Möglichkeiten, den Augen dieselben Informationen auf ganz anderen Wegen zukommen zu lassen – und so das Gehirn zu täuschen. Da wären z.B. diese 2-farbigen Brillen. Oder Geräte, mit denen jedem Auge ein Bild gezeigt wird, daß dieselbe Szene darstellt, aber um den Abstand der Augen verschoben. Eben so, wie wir sie sehen würden, wenn wir wirklich da wären. (Macht doch mal Dias auf die Art, und spielt ein bisschen rum. Ein Dia-Betrachter vor jedes Auge halten, mit je einem Dia von derselben Szene, aber mit kleinen Änderungen, z.B. Abstand der Kamera bei den beiden Bildern variieren. Oder noch besser: Linkes und rechtes Bild vertauschen!)

Die Methoden gehen immer auf dasselbe Prinzip zurück: Das, welches wir oben zuerst betrachtet haben und das auch in Skizze 1 beschrieben ist. Die anderen beiden Merkmale werden vom Gehirn also offensichtlich als untergeordnet betrachtet und im Zweifelsfall – wenn sie widersprüchliche Informationen liefern – bei der Konstruktion der 3-dimensionalen Vorstellung nicht genutzt bzw. ausgefiltert.

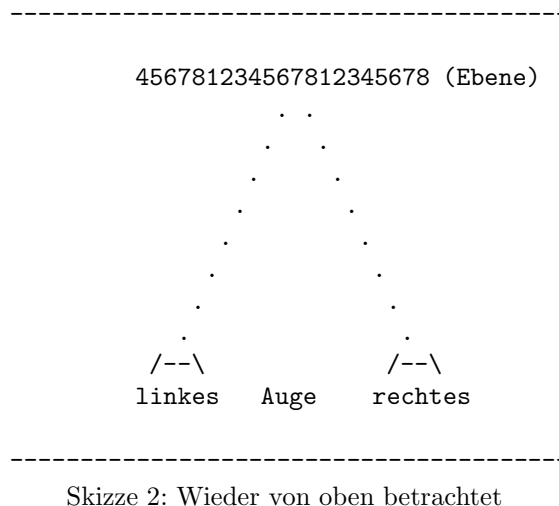
2 Chaos-Grafiken

2.1 Ansatz

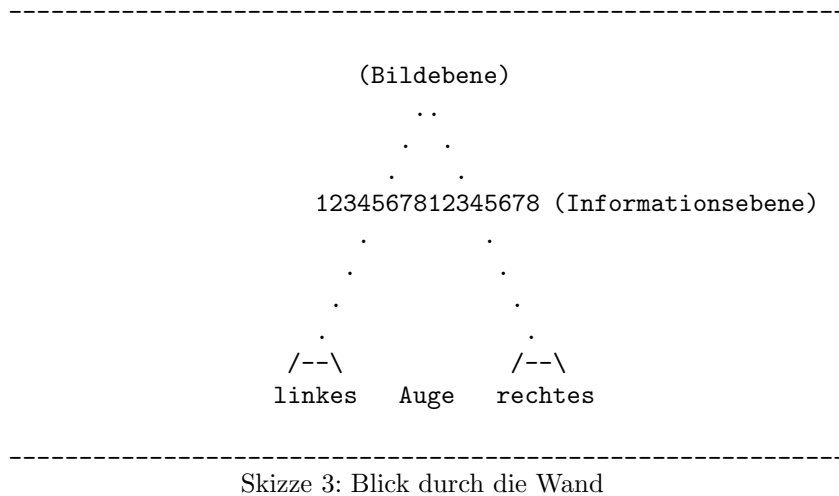
Wesentlich für die Wahrnehmung 3-dimensionaler Bilder ist also, daß beide Augen leicht unterschiedliche Informationen erhalten, die das Gehirn zu einem Bild vereinigen kann.

Hier setzen wir nun mit unserer Kodierungsmethode an: Wer schreibt uns denn eigentlich vor, daß die Muster, die wir den Augen zu fressen geben, realistisch sein müssen? Wenn das, was wir bisher gesagt haben, wahr ist, können wir die Informationen auf jede beliebige Art verpacken. Einzig wichtig ist, daß die Augen leicht verschiedene Informationen erhalten. Das geht zum Beispiel so:

Wir betrachten ein Blatt, auf dem sich ein wiederholendes Muster steht (Skizze 2). Wenn wir 'normal' auf dieses Blatt sehen, erhält unser Gehirn alle seine Informationen in der gewohnten Weise und sagt: „Ja, da ist eine Ebene, auf der stehen Zeichen, die sich wiederholen.“



Nun machen wir mal etwas anderes: Wir sehen durch das Blatt hindurch (auf den Punkt, der in Skizze 3 Bildebene genannt ist), und zwar so, daß beide Augen auf verschiedene Teile des Musters (Informationsebene in Skizze 3) sehen. Diese Teile sind aber, da sich das Muster wiederholt, identisch. Im Gehirn überlagern sich diese beiden Muster zu einem einzigen. Was leicht ist, da es keine Unterschiede zwischen den beiden Mustern gibt. Das Gehirn sagt sich wieder: „Oh, noch eine Ebene mit einem Muster. Aber Moment, da ist doch etwas anders.“ Wir sehen wieder eine Ebene, haben aber leichte Mühe, ihre Entfernung zu bestimmen. Die Informationen, die wir durch Blickwinkel etc. erhalten, sind also doch nicht ganz unwichtig. Mir wird halt immer etwas schwindelig beim Hinsehen. Aber das wichtigste: es klappt!



Bevor es so richtig losgeht, fassen wir noch einmal zusammen: Wir sehen durch das Bild, und beiden Augen sehen verschiedene Stellen des Textes. Da dieser aus einer sich wiederholenden Folge von Buchstaben besteht, erhalten die Augen dieselben Informationen und das Gehirn konstruiert daraus die Vorstellung, daß es auf eine Ebene sieht. Im folgenden nenne ich die Anzahl der unterschiedlichen Zeichen in einem Muster seine Frequenz. Das Muster in den Skizzen 1 – 3 hat also eine Frequenz von 8.

2.2 Through the Looking Glass

Wenn wir jetzt mal die Theorie für einen Moment hinter uns lassen und uns der Praxis zuwenden, sieht das so aus (3D-Bild 1):

- Wir haben ein sich wiederholendes Muster.
- Die beiden Augen betrachten verschiedene Stellen.
- Wir sehen wieder dasselbe Muster (mit etwas Schwindelgefühl dabei).

Die beiden 'Y' dienen dabei als Peilhilfe. So wird es einfacher, die Muster richtig zu überlagern. Richtig ist es, wenn drei 'Y' zu sehen sind. (Übung zum Verständnis: Wieso ist das so?) Daß ich mehrere identische Muster untereinander schreibe, dient ebenfalls nur als Sehhilfe. In der Tat könnten wir mit etwas Übung in nur einer Zeile dasselbe sehen.

```

-----
                Y      Y
1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
-----

```

3D-Bild 1: Das erste 3D-Bild

Auf diese Art sieht man ein Bild, das es so eigentlich nicht gibt. Da beide Augen in unserem Beispiel dasselbe sehen, ist das Ergebnis langweilig. Aber in der Tat ist es das einfachste drei-dimensionale Bild – eine Ebene.

2.3 Das erste echte Bild

Jetzt wollen wir mal etwas mit dieser Ebene spielen: Wie oben gesehen, betrachten die Augen nicht dieselbe Stelle des Textes. Dadurch sind nette Manipulationen möglich. Was ist, wenn wir einfach mal ein Zeichen in jeder Zeile löschen (3D-Bild 2: die '7' unter dem rechten 'Y' in den mittleren 3 Zeilen)?

```

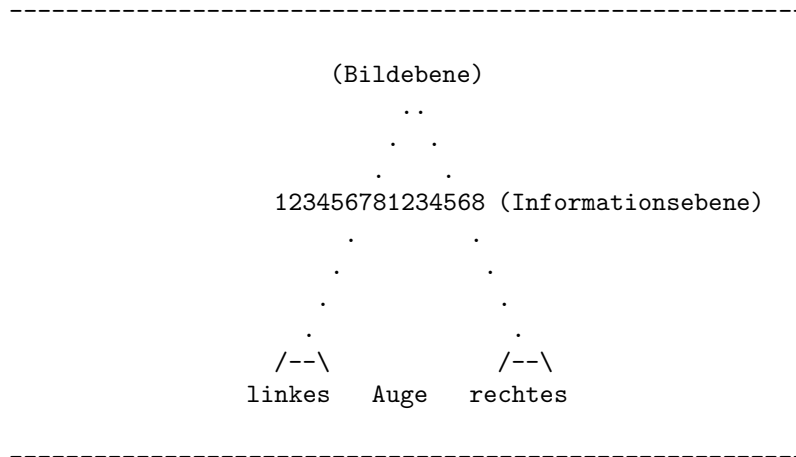
-----
                Y      Y
1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
-----

```

3D-Bild 2: Ein „echtes“ 3D-Bild

Da die Augen, wenn sie die entsprechenden Stellen (bedenkt: es sind deren 2!) betrachten, nehmen sie geringfügig unterschiedliche Informationen wahr. Wie oben gesehen, sagt sich das Gehirn in einem solchen Fall, daß der Grund wohl darin liegt, daß die Augen etwas plastisches wahrnehmen. Beim Löschen eben etwas, das vor der Ebene liegt. Wie man bemerkt, bewirkt das Löschen eines Zeichens das Verschieben eines ganzen Blockes in der Größe der Frequenz des Musters, vermindert um 1.

Wir wollen mal versuchen, den Grund dafür zu finden. Der Hintergrund (die eigentliche Ebene) hat eine Frequenz von 8. Durch das Löschen entsteht ein kleiner Bereich mit einer Frequenz von 7. (Skizze 4: die '7' im rechten Bereich ist gelöscht. Wie gesehen, tritt der Bereich von 8...6 hervor.)

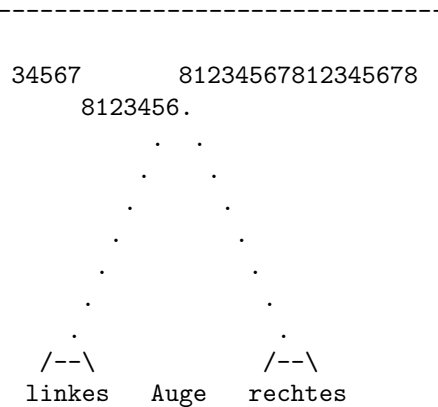


Skizze 4: Ein Zeichen wurde gelöscht

Es ist ja ein bisschen vermessen zu vermuten, wie das Gehirn aus der optischen Wahrnehmung eine 3-dimensionale Vorstellung der Welt konstruiert, aber ich erkläre mir das so: (Skizze 4 zeigt einen Ausschnitt aus 3D-Bild 2. Der Bereich liegt 3 Zeilen unter dem rechten 'Y')

- Das Gehirn registriert eine Änderung des Musters. Dabei sieht das rechte Auge das Muster mit der 7, das linke nicht.
- Wie oben gesehen (Abschnitt 1), liefern die unterschiedlichen Informationen, die bei beiden Augen erhalten, wichtige Anhaltspunkte für die 3-dimensionale Vorstellung, die wir uns über die Welt machen.
- Wenn das linke Auge etwas sieht, was das rechte nicht sehen kann, nimmt das Gehirn an, daß ein Objekt so liegt, daß es den für das eine Auge unsichtbaren Teil verdeckt. Das geht nur, wenn sich das Objekt vor dem verdeckten Teil befindet. (vergl. Skizze 1)
- Das Gehirn denkt: „Verstanden, da ist etwas vor der '7'.“ Und schwupps, erscheint uns das Muster nach vorne verschoben.
- Mit demselben Gedanken wird klar, daß dieser Bereich nur bis zur nächsten '7' reicht. Wie man leicht sehen kann (3D-Bild 2), erhalten die Augen wieder identische Informationen, wenn man die Bereiche betrachtet, die sich links und rechts anschließen. (Würden wir auch im folgenden Muster die '7' löschen, würde sich der vortretende Bereich entsprechend verlängern.)

Das ist es also, was wir uns vorstellen:

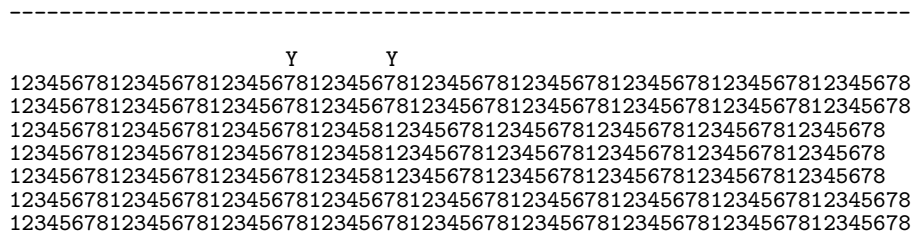


Skizze 5: Unsere Vorstellung

Wir haben hier nur bedacht, daß das rechte Auge die Änderung des Musters sieht und überlegt, wie das Gehirn dies interpretiert. Genauso läßt sich auch die Änderung und die daraus resultierende Vorstellung erklären, wenn das linke Auge auf den Ort der Änderung blickt. (Versucht selber, die Erklärung zu geben.)

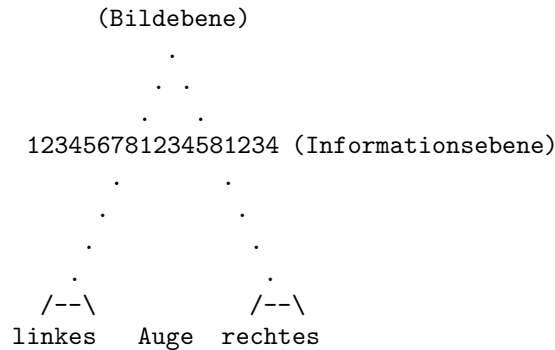
2.4 Größere Abstände

Mal sehen, was geschieht, wenn wir mehrere Zeichen löschen. Also werfen wir nun auch die '6' heraus (3D-Bild 3):



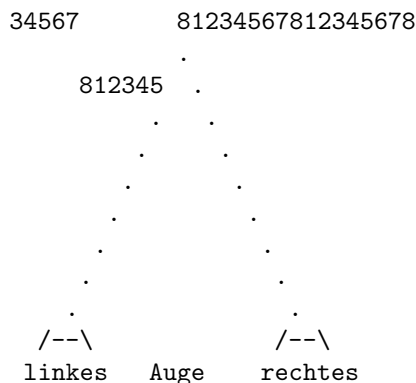
3D-Bild 3: Zwei Zeichen wurden gelöscht

Wir sehen jetzt auf diese Muster:



Skizze 6: Zwei Zeichen wurden gelöscht

Daraus wird offensichtlich dies:



Skizze 7: Unsere Vorstellung

Am Prinzip hat sich nichts geändert.

Die Erklärung, die das Gehirn für diese Änderung findet, ist nun, daß das 'Objekt', das für die Änderung verantwortlich ist, weiter vorne liegt und daher einen breiteren Bereich verdeckt. Da wir einen Mindestabstand haben, um den das Muster verkürzt wird (die Zeichenbreite), erfolgt die Konstruktion, wo die Objekte liegen, immer sprunghaft. Es gibt Ebenen, auf denen die Objekte liegen können. Eine feinere Abstufung kann erreicht werden, indem die Breite eines Zeichens verkleinert wird (grafische Ausgabe, anderer Zeichensatz, Verkleinerung im Copy-Shop etc).

2.5 Die andere Richtung

So, wie wir einen Bereich nach vorn verschoben erscheinen lassen, wissen wir jetzt. Als nächstes wollen wir etwas nach hinten treten lassen. Dazu nehmen wir uns wieder ein Muster und – naja, nach vorn verschieben erreichen wir durch

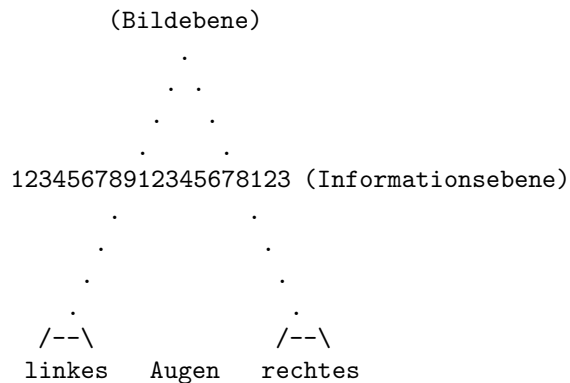
Löschen, dann wird's nach hinten wohl durch Einfügen gehen. Das ist auch so, wie 3D-Bild 4 zeigt. Wieder ist das rechte 'Y' der Ort, an der sich das Muster ändert. Die Argumentation erfolgt analog zur eben gegebenen, daher werden wir nun etwas schneller vorangehen.

Y Y

1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
12345678123456781234567812345678912345678123456781234567812345678
12345678123456781234567812345678912345678123456781234567812345678
12345678123456781234567812345678912345678123456781234567812345678
12345678123456781234567812345678912345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678

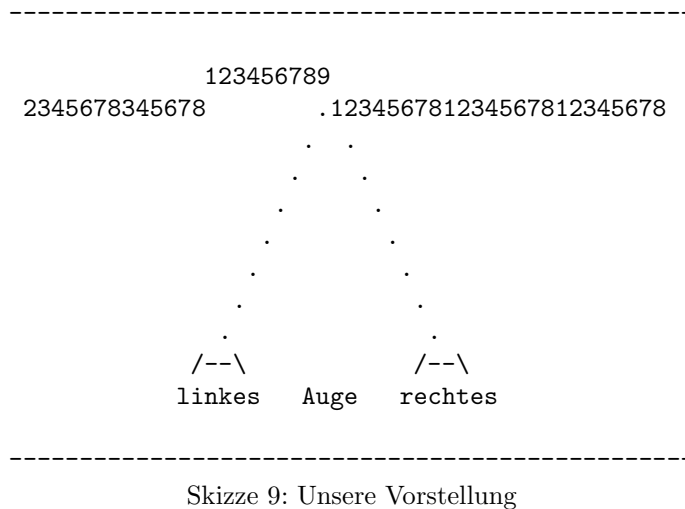
3D-Bild 4: Ein Zeichen wurde eingefügt

Wie gehabt nehmen wir eine Änderung des Musters wahr. Nur sieht diesmal, wenn wir die geänderte Stelle betrachten, das linke Auge die '9', das rechte aber nicht (Skizze 8).



Skizze 8: Ein Zeichen wurde eingefügt

Und das macht das Gehirn daraus (Skizze 9): (Wer genau hinsieht, stellt fest, daß die '9' nicht eindeutig einem Bereich zuzuordnen ist. Mir erscheint sie 'zwischen' vorn und hinten.



Auch hier gilt wieder: wenn wir eine weitere '9' an der richtigen Stelle im Muster einfügen, erscheint ein breiterer Bereich verschoben.

2.6 Zusammenfassung 1

Wir können ganz allgemein so argumentieren: Wir haben ein Muster mit einer Frequenz n . Löschen wir ein Zeichen im Muster, erscheint ein Bereich der Länge $n-1$ um eine Einheit nach vorn verschoben. Fügen wir ein Zeichen ein, tritt ein Bereich der Länge $n+1$ eine Ebene zurück. Der Bereich wird breiter, wenn wir die Änderung im benachbarten Muster übernehmen. Natürlich können wir durch Einfügen bzw Löschen mehrerer Zeichen den Bereich auch um mehrere Ebenen verschieben. Das Prinzip bleibt dasselbe.

Oder anders gesagt:

Durch die Änderung der Frequenz des Musters erscheint der Bereich verschoben, in dem diese Änderung enthalten ist - unabhängig davon, wie breit er ist. Eine Verkleinerung der Frequenz bewirkt eine Verschiebung zum Betrachter hin, eine Vergrößerung verschiebt das Muster von ihm weg. Die Größe der Änderung bestimmt, um wie viele Ebenen der Bereich verschoben wird. Dies gilt auch für Bereiche, die nicht exakt der Frequenz des Musters entsprechen – also auch für solche, die kürzer als die Frequenz sind! Das wird uns den Weg bereiten, 3-dimensionale Grafiken zu realisieren.

2.7 Kürzere Muster

Nachdem wir nun die grundlegende Technik verstanden haben, wollen wir uns überlegen, wie wir kleinere Bereiche verschieben können. Eben haben wir gesehen, daß wir die Länge des Bereiches variieren können. Wieso sollte das nur für lange Stücke gelten? Wer sagt uns denn, daß dasselbe Prinzip nicht auch bei kurzen Textstücken klappt? Probieren wir das aus. Wir nehmen uns ein Muster mit einer Frequenz von 8 und wollen – sagen wir mal – einen Bereich der Länge 3 um eine Ebene nach vorn treten lassen. Dazu löschen wir zuerst einmal ein Zeichen im Muster. Dann wird ein Bereich von 7 Zeichen nach vorn treten. Da

uns das aber zu viel ist, fügen wir nach 3 Stellen wieder ein Zeichen ein. Hier ein 'X'. Mal sehen, was dann geschieht...

```

-----
                Y      Y
1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
123456781234567812345678123X456812345678123456781234567812345678
123456781234567812345678123X456812345678123456781234567812345678
123456781234567812345678123X456812345678123456781234567812345678
123456781234567812345678123X456812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
-----

```

3D-Bild 5: Ein Versuch...

Scheint ja zu klappen. Aber nur beinahe. Es ist uns zwar gelungen, einen Bereich nach vorn zu verschieben, genau wie wir es wollten. Die Idee ist also offensichtlich richtig. Gleichzeitig ist aber links von der betrachteten Stelle ein anderer Bereich nach hinten getreten. Wir haben nämlich etwas wichtiges übersehen. Wir haben ein 'X' in das Muster eingefügt - und damit das Muster verlängert! Wie wir wissen, bewirkt das eine Verschiebung nach hinten. Damit haben wir – sozusagen en passant – gezeigt, daß unsere Idee in beiden Richtungen funktioniert. Aber eben auch, daß sie noch verbesserungsfähig ist.

Diese Verbesserung ist aber einfach. Wir müssen das Muster eben so anpassen, daß es sich wieder wiederholt, und zwar mit der Frequenz, die wir haben wollen. Also müssen wir die Änderung, die durch das Einfügen eines Zeichens entstanden sind, einfach in den Nachbarmustern übernehmen. Aus dem Muster '12345678' wird also, bedingt durch die Stelle, an denen wir ein Zeichen eingefügt haben, das Muster '123X4568'. Jetzt sollte es richtig funktionieren. 3D-Bild 6 zeigt, daß dem auch so ist.

```

-----
                Y      Y
1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
123X4568123X4568123X4568123X456812345678123456781234567812345678
123X4568123X4568123X4568123X456812345678123456781234567812345678
123X4568123X4568123X4568123X456812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
1234567812345678123456781234567812345678123456781234567812345678
-----

```

3D-Bild 6: ... und die Vollendung

(Daß ich das Muster von rechts nach links ändere, ist allein darauf zurückzuführen, daß ich in diesem Text eine Änderung des Musters immer von rechts aus betrachtet habe - ich bin halt Rechtsäuger. Die Änderung kann aber problemlos auch anders herum gedacht und vorgenommen werden. In der Tat habe ich im Programm, weil's zu nervig wurde, immer 'falschrum' zu denken, schließlich alle Indizes von links nach rechts laufen lassen. Abschnitt 3 geht zur Abwechslung mal richtigrum vor.)

2.8 Zusammenfassung 2

So! Nun wissen wir, wie wir Bereiche vor- und zurücktreten lassen können, und zwar in jeder Breite und um beliebig viele Ebenen. Was wir jetzt noch benötigen, ist die praktische Umsetzung für das, was wir gerade mehr experimentell und umgangssprachlich geklärt haben. Und das geht so:

3 Realisierung

Wir wollen die folgende Figur zu Fuß erzeugen:

```

-----
      oooooooooo  oooooooooo
      ooo
-----
Skizze 10: Hausaufgabe

```

Die Figur liegt 2 Ebenen vor dem Hintergrund.
Frequenz = 8, Muster = '12345678':

```

-----
      A  B
1234567812  cd678345cd678
      .  .
      . 345 .
      .  .
      .  .
      /-\  /-\
-----
Skizze 11: Das wollen wir sehen

```

An den Orten der Änderung geschieht folgendes:

- A) Bereich tritt vor: (Rechtes Auge kann '12' nicht sehen, weil diese Zeichen von der Figur verdeckt werden. Vergl. Abschnitt 1.)
- Muster wird ab '1' um 2 Zeichen nach linke verschoben.
-> '12' wird **gelöscht**.
 - Frequenz = 6.

Muster: '345678'

- B) Bereich tritt (wieder) zurück: (Linkes Auge kann 'cd' nicht sehen. Begründung wie oben.)

- Frequenz = 8.
- Muster wird ab '6' um 2 Zeichen nach rechts **verschoben**
-> Es entsteht Platz für 2 neue Zeichen.
- In die entstandene Lücke wird 'cd' **eingefügt**.

neues Muster: '345cd678'

Wenn die Figur noch weiter von der Hintergrundebene entfernt liegt, werden entsprechend mehr Zeichen verdeckt. Logisch...

Das Ergebnis zeigt 3D-Bild 7:

```

-----
                                Y      Y
1234567812345678123456781234567812345678123456781234
1234567812345678123456781234567812345678123456781234
1234567812345678123456781234567812345678123456781234
1234567812345678123456781234567812345678123456781234
1234567812345678123456781234567812345678123456781234
123456781234567812345678345cd8ab345cd8ab345cd8ab345cd8ab345c
123456781234567812345678345cd8ab345cd8ab345cd8ab345cd8ab345c
123456781234567812345678345cd8ab345cd8ab345cd8ab345cd8ab345c
123456781234567812345678345cd8ab345cd8ab345cd8ab345cd8ab345c
123456781234567812345678345cd8ab345cd8ab345cd8ab345cd8ab345c
123456781234567812345678345cd8ab345cd8ab345cd8ab345cd8ab345c
123456781234567812345678345cd8ab345cd8ab345cd8ab345cd8ab345c
123456781234567812345678123456781234567812345678123456781234
123456781234567812345678123456781234567812345678123456781234
123456781234567812345678123456781234567812345678123456781234
-----

```

3D-Bild 7: Fertig

Damit ist das Prinzip der wichtigsten Routine des Programms formuliert. Wir wissen, wie wir das Hintergrundmuster verändern müssen, um dem Gehirn den gewünschten Effekt vorzumachen.

3.1 Die Mathematik der Chaos-Grafik

Wie wir gesehen haben sind dies die Faktoren, die bei der Beschreibung einer 3-D Grafik eine Rolle spielen:

Grundfrequenz: 8

d: Abstand des Punktes zum Hintergrund

f: Frequenz des Musters

Das brauchen wir außerdem:

i: fortlaufender Index.

g-i: gewünschter Index (zeigt auf das Zeichen im Muster, daß wir als nächstes brauchen)

P_A: Muster-Anpassung

Und so hängen die Faktoren zusammen:

d:	0	0	0	0	0	0	-1	-1	-1	-1	-1	-1	-1	0	0	0	0	0	0	0	0	
f:	8	8	8	8	8	8	7	7	7	7	7	7	7	7	8	8	8	8	8	8	8	
i:	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P_A:							X							X								
g_i:	4	3	2	1	0	8	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0

Wie erhalten wir nun g_i? Versuchen wir's erst einmal auf deutsch:

Aus dem Textfile wird die Information über den nächsten Grafik-Punkt geholt:

d = Abstand Punkt / Hintergrund

Wenn der Abstand sich (bezogen auf den vorherigen Punkt) geändert hat:

- wird das Chaos-Muster geändert:
Ein Teil wird verschoben,
ggf. werden neue Zeichen erzeugt.
- Neue Frequenz berechnen:
 $f = \text{standart_frequenz} + d$
g_i an frequenz anpassen:
 $g_i = g_i \bmod f$

Den Wert des Chaos-Musters an nächster Grafik_Position speichern

Nächsten Index berechnen, mit Anpassung an Frequenz: $g_i = (g_i + 1) \bmod f$

An den markierten Stellen muß das Muster geändert werden. (Warum und wie? Noch einmal Abschnitt 3.1 lesen!)

4 Chaos(–Grafik) in C

Jajaja, es ginge auch alles kürzer, trickreicher und ... unverständlicher. Aber es soll jeder kapieren. Das ist schließlich der Grund, warum ich Programm und Text geschrieben habe. Naja, egal. Das, worüber sich die 'echten' C-Programmierer aufregen können, liegt als Source-Code unter dem Namen DDD.C bei. Zusammen mit dem File TOOLS.C (Kommando-Zeilen- Krempel und TOS-abhängige IO-Routinen) kann das Programm erzeugt werden. Sollte mit so ziemlich jedem Compiler aus diesem Jahrhundert übersetzt werden können. Ich hab's geschrieben mit Turbo-C 2.0 auf dem Atari ST. Alle benötigten Sourcen und das Projekt-File liegen bei.

5 Schluss

Der, wie immer, an den Anfang anknüpft. Es **gibt** verschiedene Wege, das Prinzip zu erklären. Hier ist ein recht einfacher vorgestellt worden. Vor kurzem ist ein sehr guter Artikel samt Programm in der c't vorgestellt worden (Heft 7, 1994, S. 230f). Wer es also noch genauer wissen möchte oder einen PC hat und richtige 3-D Grafiken machen möchte, sollte dort weiterlesen. Es gibt an dieser Stelle auch eine kurze Literaturliste zum Thema.