Mathematik erzeugt grafische Kunstwerke und zauberhafte Videos:

Was sind Fraktale?

Klaus Kusche

Frühjahr 2019

Inhalt

- Unser Ziel
- Was ist ein Fraktal?
- Von linearen geometrischen Abbildungen zu iterierten Funktionssystemen (IFS-Fraktalen)
- Berechnung von Mandelbrot- und Julia-Fraktalen (“Apfelmännchen”)

Ziel

“<u>Rein künstliche</u>”
mathematische erzeugte Bilder

==> *<u>Kein reales Bild</u>* und keine Zeichnung
als Ausgangspunkt

==> *<u>Keine geometrische Modellierung</u>*
realer Welten

... mit *<u>relativ einfachen Mitteln</u>*
(~ 10. Klasse Mathe, 1 Jahr gut Proggen gelernt)

***“Belohnungs-Übungen”* im 1. Jahr BK Informatik**

Mathematik oder Informatik?

Reine Mathematik !

Der Computer ist nur *“notwendiges Hilfsmittel”*:

Dieselben einfachen Berechnungen
werden ***viele Millionen Mal wiederholt***
(Apfelmännchen wurden vor 30 Jahren sogar als Benchmark verwendet!)

==> Ohne Computer ginge es nicht!

“Einfache” Mathematik
+ clevere Tricks & Ideen
+ viel Rechenkraft
= schöne Bilder

Was sind Fraktale? (1)

Fraktale sind
<u>unendlich fein strukturierte</u>
bzw. "zerstückelte" Gebilde!

==> Man kann zumindest an einigen Stellen
(bei manchen Fraktalen auch an allen!)

<u>beliebig tief</u> hineinzoomen

und bekommt

immer wieder <u>neue Details</u>

zu sehen.

Was sind Fraktale? (2)

Fraktale sind
"<u>selbstähnliche</u>" Gebilde!

==> Beim <u>*Hineinzoomen*</u>
sieht man immer wieder

<u>verkleinerte Kopien</u>
derselben Strukuren

(entweder <u>*ident*</u> oder <u>*leicht verändert*</u>).

==> *Viele Fraktale "<u>enthalten sich selbst</u>"!*

Einfache Fraktale

In der Natur:

- *Schneeflocken*
- *Bäume* und manche andere Pflanzen

In Mathe: "Geometrische" Fraktale

- *Koch'sche Schneeflocke, Hilbert-Kurve, ...*

  ==> *Unendlich lange Linie* in *endlicher Fläche*!
- *Sierpinski-Dreieck, Pythagoras-Baum, ...*

  ==> Immer wieder *dieselbe geom. Konstruktion*

“Komplizierte” Fraktale

“***Kontinuierliche***” Berechnungen statt geometrischer Konstruktionen

Mehrere *völlig unterschiedliche* Arten:

- *Iterierte Funktionssysteme (IFS)*
- *Mandelbrot- und Julia-Mengen (“Apfelmännchen”)*
- *“Chaotische Attraktoren”, z.B. Lorenz-Attraktor*

  (eigentlich *keine* Fraktale!)
  (extrem *schwierige* Mathematik!)
- … und einige andere mehr!

Anwendung von Fraktalen

- Wichtiger Teil der *Computerkunst*
- Auch in Programmen,
  die *künstliche Landschaften* usw. generieren

Jahrzehnte altes Highlight:

- ***"Electric Sheep"*** bzw. ***"Fractal Flame"***
- *Screensaver* mit weltweit verteiltem,
  kooperativem Rechen-Netz (wie SETI@home)
- Kombiniert *erweiterte IFS* & *Nachbearbeitung*

Lineare geometrische Abbildung

= Koordinatentransformation

= bildet Punkte aus einem Bereich der Ebene
<u>in einen anderen Bereich</u> ab

$$x_{neu} = c_1 * x_{alt} + c_2 * y_{alt} + c_3$$

$$y_{neu} = c_4 * x_{alt} + c_5 * y_{alt} + c_6$$

Je nach $c_1 \ldots c_6$ (vorgegeben):

Spiegeln, verschieben, vergrößern/verkleinern, rotieren, stauchen/strecken, scheren, …

Iterierte Funktionssysteme (1)

*Iteriert heißt “**wiederholt**”:*

Die Transformation wird angewendet

- nicht einmal auf ein ganzes Bild
- sondern *hunderttausende Mal nacheinander* auf einen *einzelnen Punkt*

==> liefert ein *Bild bestehend aus vielen Punkten*!

(aber noch nicht wirklich spannend: Spiralen usw.)

Iterierte Funktionssysteme (2)

"Funktions<u>systeme</u>": Der geniale Trick für

"<u>abwechslungsreiche</u>" Bilder

- *<u>Nicht eine</u>* Transformation ...
- ... sondern *<u>mehrere</u>* verschiedene (2 ... ~20)

In jedem Schritt:

Wähle <u>zufällig</u> eine der Transformationen
(mit verschiedenen Wahrscheinlichkeiten)

==> *"Würfle und verschiebe dann den Punkt dahin oder dorthin oder ganz woanders hin"*

Beispiel IFS-Farn:
4 Transformationen

- *W = 85 %: Leicht verkleinern, leicht nach rechts drehen, ein Stück nach schräg rechts oben verschieben*

  ==> erzeugt die *nächste Blattreihe aus der darunter*

- *W = 7 %: Stark verkleinern, rund 90° links drehen, nach ganz links unten verschieben*

  W = 7 %: Stark verkleinern, rund 90° rechts drehen, spiegeln, nach ganz rechts unten verschieben

  ==> erzeugen *linkes & rechtes unterstes Blatt*

- *W = 1 %: Auf kurzen senkrechten Strich ganz unten in der Mitte komprimieren*

  ==> erzeugt *unterstes Stück vom Stamm*

Einfache Erweiterungen

- *Koeffizienten* $c_1 \dots c_6$ nach jedem Bild *leicht ändern*

  ==> ***bewegte Bilder***

- Zu jeder Koordinaten-Transformation auch je eine *Farb-Transformation* definieren

  (d.h. berechne rot_{neu}, grün_{neu} und blau_{neu} aus rot_{alt}, grün_{alt} und blau_{alt})

  ==> ***bunte Bilder, Farbverläufe***

Mandelbrot & Julia (1)

"Berechne der Reihe nach für <u>jedes Pixel</u> der Bildfläche eine <u>Farbe</u>"

Bildfläche:

Ausschnitt der <u>*reellen Ebene*</u>
in der <u>*Nähe*</u> von 0/0

Gegeben: <u>***Pixelkoordinaten***</u> p_x / p_y

($-2 <= p_x <= 2$ und $-2 <= p_y <= 2$,
alle Punkte weiter <u>*außen*</u> haben immer <u>*dieselbe Farbe*</u>)

Gesucht: <u>*Farbnummer*</u> *im Farbkreis (zyklisch)*

Mandelbrot & Julia (2)

Zusätzlich gegeben:

Zwei <u>Konstanten</u> c_x / c_y

- Für alle Pixel *<u>gleich</u>*
- Im einfachsten Fall 0 / 0,
  jedenfalls auch "klein" (< 2)
- *"<u>Stellschraube</u>"*:
  Ändern bzw. verzerren das Bild

Mandelbrot & Julia: Formel

Mandelbrot:

$$x_0 = c_x$$
$$y_0 = c_y$$
$$x_{n+1} = x_n^2 - y_n^2 + p_x$$
$$y_{n+1} = 2 * x_n * y_n + p_y$$

Julia:

$$x_0 = p_x$$
$$y_0 = p_y$$
$$x_{n+1} = x_n^2 - y_n^2 + c_x$$
$$y_{n+1} = 2 * x_n * y_n + c_y$$

(es gibt auch Formeln für *höhere Potenzen*)

Wesentlicher **Unterschied zu IFS**:

***Nichtlineare* Formeln!!!**

(enthalten Produkte & Potenzen von Variablen!)

Mandelbrot & Julia: Berechnung

Berechne immer wieder das nächste x_n / y_n

- bis entweder $x_n^2 + y_n^2 >= 4$ ist

  (d.h. x_n / y_n liegt *außerhalb des Kreises* mit r = 2)

  ==> **n ist die gesuchte Farbnummer für p_x / p_y!**

- oder n die *vorgegebene Maximal-Anzahl* erreicht hat (typischerweise 100 … 1000):

  ==> **Mach das Pixel p_x / p_y schwarz**

  ("Innenbereich", Hintergrundfarbe)

Mandelbrot & Julia: Anschaulich

<u>Wie viele Punkte</u> x_n / y_n kannst du ausgehend von p_x / p_y berechnen, bis du aus dem Kreis mit r = 2 <u>herausfällst</u>?

<u>Anzahl der Punkte</u> = *<u>Farbnummer</u>* von p_x / p_y

Fällt *<u>gar nicht</u>* heraus = p_x / p_y ist *<u>schwarz</u>*

Aber: ***Die Berechnung verhält sich***

<u>extrem instabil</u> (chaotisch)!

==> *<u>Eng beisammenliegende</u>* p_x / p_y ergeben oft *<u>völlig unterschiedliche</u>* x_n / y_n !!!

Mandelbrot & Julia: Erweiterung

Alternative Farbgebung statt Anzahl der x_n / y_n:

Winkel vom Punkt 0 / 0
zum *ersten* x_n / y_n *außerhalb* r = 2
bestimmt die Farbe

(z.B. = Winkel im Farbkreis)

Chaos in sonst schön *strukturierten Bereichen*
Schöne Muster in sonst einfärbig *faden Bereichen*

Und für ***Bewegung***: *Hineinzoomen!*

“The end”

Fragen?