





Universität Passau  
Fakultät für Mathematik mit Schwerpunkt Digitale  
Bildverarbeitung

## Bachelorarbeit

im Studiengang Informatik

zur Erlangung des akademischen Grades  
Bachelor of Science

- Thema:** Erzeugung und Darstellung von Fraktalen  
mithilfe von Iterierten Funktionensystemen
- Autor:** Martin Schütz  
MatNr. 67242
- Version vom:** 9. Oktober 2015
- 1. Betreuer:** Prof. Dr. Sauer  
**2. Betreuer:** Dipl.-Inf. Stefan Schröder

## Abstract

Eine wichtige Aufgabe der Naturwissenschaften ist es, in der Natur auftretende Objekte möglichst genau zu beschreiben. Für viele dieser Objekte reicht die klassische Geometrie aus, allerdings existieren auch Dinge, wie zum Beispiel der Blumenkohl Romanesco, deren Form sich aufgrund ihrer detailreichen Oberfläche nicht mit den Formen aus der klassischen Geometrie beschreiben lässt. An dieser Stelle setzt die fraktale Geometrie ein. Sie ist in der Lage, Objekte zu beschreiben, die eine unendlich detaillierte Form besitzen. Mithilfe der im Folgenden vorgestellten Theorie lassen sich erstaunliche Objekte erzeugen. Dabei werden zunächst die mathematischen Grundlagen für die Erzeugung von Fraktalen gelegt und anschließend wird eine konkrete Implementierung dafür vorgestellt. Zuletzt wird ein Ausblick auf die Möglichkeiten der fraktalen Geometrie gegeben.

# Inhaltsverzeichnis

<b>Abbildungsverzeichnis</b>	<b>5</b>
<b>Tabellenverzeichnis</b>	<b>5</b>
<b>1 Einleitung</b>	<b>6</b>
<b>2 Mathematische Theorie zu Fraktalen</b>	<b>7</b>
2.1 Definition von Fraktalen . . . . .	7
2.1.1 Der metrische Raum $(\mathcal{H}(\mathbf{X}), h)$ . . . . .	7
2.1.2 Kontraktionen . . . . .	8
2.1.3 Iterierte Funktionensysteme . . . . .	9
2.2 Der deterministische Algorithmus . . . . .	10
<b>3 Eigenschaften von Iterierten Funktionensystemen und Anforderungen an deren Transformationen</b>	<b>11</b>
3.1 Skalierung von Fraktalen . . . . .	11
3.2 Bestimmung einer geeigneten Detailtiefe . . . . .	12
3.3 Selbstähnlichkeit . . . . .	12
3.4 Anforderungen an die Matrizen der Transformationen . . . . .	12
3.4.1 Kontraktionen im $\mathbb{R}^2$ . . . . .	13
3.4.2 Flächeninhalt von Fraktalen . . . . .	13
<b>4 Programm zur Darstellung von Fraktalen</b>	<b>16</b>
4.1 Datenstrukturen . . . . .	16
4.2 Implementierung des Algorithmus . . . . .	16
4.3 Graphische Darstellung . . . . .	17
4.3.1 Die 3D-Darstellung des Fraktals . . . . .	17
4.3.2 Der Eingabebereich . . . . .	17
4.4 Beispiele von erzeugten Fraktalen . . . . .	17
4.4.1 Das Sierpinski-Dreieck im zwei- und dreidimensionalen Raum .	18
4.4.2 Der Menger-Schwamm . . . . .	19
4.4.3 Der Barnsley-Farn . . . . .	20
<b>5 Fazit</b>	<b>21</b>
<b>Literaturverzeichnis</b>	<b>22</b>
<b>Eidesstattliche Erklärung</b>	<b>22</b>

## Abbildungsverzeichnis

1	Ein Iterationsschritt zur Erzeugung eines Sierpinski-Dreiecks . . . . .	18
2	Ein 2D Sierpinski-Dreieck . . . . .	18
3	Ein Sierpinski-Tetraeder . . . . .	19
4	Ein Sierpinski-Quader . . . . .	19
5	Ein Barnsley-Farn . . . . .	21

## Tabellenverzeichnis

1	Ein IFS für ein 2D Sierpinski-Dreieck . . . . .	18
2	Ein IFS für ein Sierpinski-Tetraeder . . . . .	19
3	Ein IFS für den Menger-Schwamm . . . . .	20
4	Ein IFS für den Barnsley-Farn . . . . .	20

## 1 Einleitung

Die klassische Geometrie ist ein hervorragendes Werkzeug zur Beschreibung vielfältiger Formen. Durch Komposition lassen sich aus einfachen Grundformen komplexe Gebilde erzeugen. Allerdings stößt diese Geometrie an ihre Grenzen, wenn sie versucht Objekte zu beschreiben, deren Form eine unendliche Detailtiefe besitzt. Das bedeutet vereinfacht gesagt, dass man immer weiter in das Objekt "hineinzoomen" kann, ohne dass dabei irgendwann eine der Grundformen aus der klassischen Geometrie auftritt. Derartige Objekte können durch die klassische Geometrie höchstens approximiert, aber niemals exakt beschrieben werden. Diese Tatsache hatte zur Folge, dass eine neue Betrachtung der Geometrie dieser Objekte entwickelt werden musste. Die fraktale Geometrie stellt diesen neuen Zweig der Geometrie dar, der sich mit der Beschreibung solcher unendlich detailreicher Formen, sogenannter Fraktale, beschäftigt.

Während man die fraktale Geometrie vielleicht oberflächlich betrachtet als "Theorie zur Beschreibung schöner Formen" abtun könnte, verbergen sich dennoch einige nützliche praktische Anwendungsfälle hinter dieser Theorie. Die fraktale Antenne zum Beispiel findet sich heutzutage in fast jedem Handy. Dabei erwirkt die Eigenschaft, dass sich die Strukturen der Antenne immer wieder im Kleineren wiederholen, einen viel größeren Frequenzbereich, als er mit einer gewöhnlichen Antenne möglich wäre.

Als einer der Urväter der fraktalen Geometrie beschäftigte sich Mandelbrot schon 1980 mit diesem Gebiet und prägte damit eines der bekanntesten Fraktale: das Mandelbrot-Fraktal. 1991 wurden dann die Julia-Mengen, die eine Verallgemeinerung des Mandelbrot-Fraktals darstellen, von Gaston Maurice Julia und Pierre Fatou untersucht. Obwohl die Erforschung der Fraktale schon über Jahre währt und bereits einige Entdeckungen auf diesem Gebiet gemacht wurden, ist dieser mathematische Zweig noch weitestgehend unerforscht.

Aufgrund der Detailtiefe von Fraktalen kann deren Form nicht direkt beschrieben werden. Vielmehr werden Fraktale durch eine Folge von Operationen definiert, welche dann gegen ein Fraktal konvergieren. Die große Schwierigkeit dabei ist, dass es selten möglich ist, aus der Beschreibung dieser Operation direkt eine Vorstellung von dem entstehenden Fraktal zu bekommen.

Für die Aufgabe, eine Folge von Operationen immer wieder hintereinander auszuführen, eignet sich besonders ein Computerprogramm. Dieses ist zum einen in der Lage, die Berechnungen verhältnismäßig schnell durchzuführen, zum anderen kann es das Ergebnis graphisch und interaktiv darstellen. Besonders im dreidimensionalen Raum ist die Eigenschaft der Interaktivität sehr nützlich um das entstandene Fraktal genau zu untersuchen. Andererseits kann ein Algorithmus zur Erzeugung von Fraktalen nicht unendlich detailreiche Formen erzeugen, da die Ressourcen eines Computers beschränkt sind. Aus diesem Grund muss zusätzlich ein Abbruchkriterium gefunden werden, so

dass der Algorithmus terminieren kann, die charakteristischen Eigenschaften des Fraktales aber trotzdem genau genug dargestellt werden.

Im Folgenden wird zunächst ein mathematisches Verfahren zur Erzeugung von Fraktalen erläutert. Danach werden einige Eigenschaften, die die durch dieses Verfahren erzeugten Fraktale aufweisen, beschrieben. Zuletzt werden dann die Ergebnisse aus den vorherigen Kapiteln genutzt, um einen Algorithmus zu schreiben, der die oben genannten Kriterien erfüllt und sich dabei die Eigenschaften der Fraktale zunutze macht.

## 2 Mathematische Theorie zu Fraktalen

Im Folgenden bezeichnet  $\mathbf{X}$  immer einen beliebigen Raum. Zusätzlich wird häufig  $(\mathbf{X}, d)$  als ein Metrischer Raum verwendet mit einer beliebigen Metrik  $d$ . Die Begriffe Raum, Metrik und Folge werden als bekannt vorausgesetzt.

### 2.1 Definition von Fraktalen

Wesentliche Voraussetzungen zur Definition und Erzeugung von Fraktalen sind der metrische Raum  $(\mathcal{H}(\mathbf{X}), h)$ , Kontraktionen und Iterierte Funktionensysteme, die im Folgenden beschrieben werden.

#### 2.1.1 Der metrische Raum $(\mathcal{H}(\mathbf{X}), h)$

Um den Raum definieren zu können, in dem Fraktale existieren, muss zuerst der Begriff der Kompaktheit erläutert werden.

**Definition 1.** Eine Teilmenge  $S \subset \mathbf{X}$  eines metrischen Raumes  $(\mathbf{X}, d)$  ist kompakt, falls jede unendliche Folge  $(x_n)_{n=1}^{\infty}$  in  $S$  eine Teilfolge enthält, die einen Grenzwert in  $S$  besitzt. [Bar95, S 22]

Eine wichtige Eigenheit der Kompaktheit ist, dass diese im Bezug auf die Vereinigung erhalten bleibt. Deshalb gilt:

**Satz 1.** Seien  $A$  und  $B$  kompakte Mengen. Dann ist auch  $A \cup B$  eine kompakte Menge.

*Beweis.* Sei  $s$  eine unendliche Folge in  $A \cup B$ . Dann existiert ohne Beschränkung der Allgemeinheit eine unendliche Teilfolge von  $s$  in  $A$ . Diese besitzt, da  $A$  kompakt ist, eine Teilfolge mit Grenzwert in  $A$  und damit auch in  $A \cup B$ .  $\square$

Mithilfe des Begriffs der Kompaktheit lässt sich nun der Raum definieren, in dem Fraktale existieren.

**Definition 2.** Für einen metrischen Raum  $(\mathbf{X}, d)$  bezeichnet  $\mathcal{H}(\mathbf{X})$  den Raum der kompakten Teilmengen von  $\mathbf{X}$ , die nicht gleich der leeren Menge sind. [Bar95, S 33]

Um mit dem Raum  $\mathcal{H}(\mathbf{X})$  arbeiten zu können, muss zuerst eine Metrik für diesen Raum definiert werden. Dafür wird zunächst der Abstand eines einzelnen Punktes zu einer Menge des Raums beschrieben.

**Definition 3.** Es sei  $(\mathbf{X}, d)$  ein vollständiger metrischer Raum,  $x \in \mathbf{X}$  und  $B \in \mathcal{H}(\mathbf{X})$ . Dann ist

$$d(x, B) = \min \{d(x, y) : y \in B\}$$

als der Abstand des Punktes  $x$  zur Menge  $B$  definiert. [Bar95, S 33]

Daraus lässt sich der Abstand zweier Mengen zueinander definieren.

**Definition 4.** Es sei  $(\mathbf{X}, d)$  ein vollständiger metrischer Raum und  $A, B \in \mathcal{H}(\mathbf{X})$ . Dann ist

$$d(A, B) = \max \{d(x, B) : x \in A\}$$

als der Abstand der Menge  $A$  zur Menge  $B$  definiert. [Bar95, S 34]

Mit diesem Abstands-Begriff für Mengen im Raum  $\mathcal{H}(\mathbf{X})$  lässt sich nun die *Hausdorff-Metrik*  $h$  definieren.

**Definition 5.** Für einen vollständigen metrischen Raum  $(\mathbf{X}, d)$  ist der Hausdorff-Abstand zwischen zwei Punkten  $A$  und  $B$  aus  $\mathcal{H}(\mathbf{X})$  definiert durch

$$h(A, B) = \min \{d(A, B), d(B, A)\}.$$

Dabei definiert  $h$  eine Metrik auf  $\mathcal{H}(\mathbf{X})$  und  $(\mathcal{H}(\mathbf{X}), h)$  stellt einen metrischen Raum dar. [Bar95, S 37]

Damit lässt sich ein Fraktal im Raum  $\mathbf{X}$  als Punkt des metrischen Raums  $(\mathcal{H}(\mathbf{X}), h)$  bezeichnen. [Bar95, S 39] Das bedeutet, dass ein Fraktal eine kompakte Teilmenge des beschreibenden Raums  $\mathbf{X}$  ist. Fraktale lassen sich vor allem für die Fälle  $\mathbf{X} = \mathbb{R}^2$  und  $\mathbf{X} = \mathbb{R}^3$  mit einer bestimmten Auflösung sehr gut graphisch darstellen, da sie dann eine Pixel- bzw. Voxel-Menge darstellen.

### 2.1.2 Kontraktionen

Um Fraktale zu erzeugen, werden bestimmte Transformationen auf Mengen aus  $(\mathcal{H}(\mathbf{X}), h)$  wiederholt ausgeführt. Damit diese Mengen konvergieren und dadurch tatsächlich ein Fraktal entsteht, müssen diese Transformationen Kontraktionen sein.

**Definition 6.** Eine Transformation  $f : \mathbf{X} \rightarrow \mathbf{X}$  auf einem metrischen Raum  $(\mathbf{X}, d)$  heißt kontrahierend oder eine Kontraktion, falls es eine Konstante  $0 \leq s < 1$  gibt, so dass

$$d(f(x), f(y)) \leq s * d(x, y), \forall x, y \in \mathbf{X}$$

gilt. Eine solche Zahl  $s$  heißt Kontraktions-Faktor für  $f$  [Bar95, S 84]

Aus dieser Definition folgt direkt, dass es zu jeder Kontraktion einen Fixpunkt gibt, der auf sich selbst abgebildet wird und gegen den alle anderen Punkte durch wiederholtes Anwenden der Kontraktion konvergieren. Für das wiederholte Anwenden von Transformationen wird eine Kurzschreibweise eingeführt.

**Definition 7.** Sei  $f : \mathbf{X} \rightarrow \mathbf{X}$  eine Transformation. Dann ist  $f^{\circ n}(x)$  folgendermaßen definiert:

$$f^{\circ n}(x) = \begin{cases} f^{\circ(n-1)}(f(x)), & \text{falls } n > 0 \\ x, & \text{sonst} \end{cases}$$

Damit folgt dieser Satz.

**Satz 2.** Es sei  $f : \mathbf{X} \rightarrow \mathbf{X}$  eine Kontraktion auf einem vollständigen metrischen Raum  $(\mathbf{X}, d)$ . Dann besitzt  $f$  genau einen Fixpunkt  $x_f \in \mathbf{X}$ . Darüber hinaus konvergiert die Folge  $\{f^{\circ n}(x) : n = 0, 1, 2, \dots\}$  für jeden Punkt  $x \in \mathbf{X}$  gegen  $x_f$ , das heißt

$$\lim_{n \rightarrow \infty} f^{\circ n}(x) = x_f$$

für jedes  $x \in \mathbf{X}$ . [Bar95, S 85]

*Beweis.* Siehe [Bar95, S 85]. □

Das bedeutet, dass eine Kontraktion auf dem Raum  $(\mathcal{H}(\mathbf{X}), h)$  genau gegen einen Punkt, also ein Fraktal, konvergiert. Im Folgenden wird eine Möglichkeit, eine solche Kontraktion zu beschreiben, vorgestellt.

### 2.1.3 Iterierte Funktionensysteme

Während Kontraktionen auf  $(\mathbf{X}, d)$  einfach zu beschreiben sind, ist das im Raum  $(\mathcal{H}(\mathbf{X}), h)$  nicht der Fall. Deshalb muss ein Weg gefunden werden, eine Kontraktion in  $(\mathcal{H}(\mathbf{X}), h)$  aus Kontraktionen des  $(\mathbf{X}, d)$  zu konstruieren.

**Hilfssatz 1.** Es sei  $w : \mathbf{X} \rightarrow \mathbf{X}$  eine Kontraktion auf dem metrischen Raum  $(\mathbf{X}, d)$  mit Kontraktions-Faktor  $s$ . Dann ist  $w : \mathcal{H}(\mathbf{X}) \rightarrow \mathcal{H}(\mathbf{X})$  definiert durch

$$w(B) = \{w(x) : x \in B\}, \forall B \in \mathcal{H}(\mathbf{X})$$

eine Kontraktion auf  $(\mathcal{H}(\mathbf{X}), h)$  [Bar95, S 90]

*Beweis.* Siehe [Bar95, S 90] □

Damit lässt sich jede Kontraktion in  $\mathbf{X}$  auch als Kontraktion in  $\mathcal{H}(\mathbf{X})$  interpretieren. Da eine solche Kontraktion insbesondere in  $\mathbf{X}$  gegen genau einen Punkt konvergiert, konvergiert sie in  $\mathcal{H}(\mathbf{X})$  gegen die Menge mit genau diesem Punkt. Um interessantere Mengen zu erzeugen, können Kontraktionen auch in einem Iterierten Funktionensystem kombiniert werden.

**Definition 8.** Ein Iteriertes Funktionensystem(IFS) besteht aus einem metrischen Raum  $(\mathbf{X}, d)$  und einer endlichen Menge von Kontraktionen  $w_n : \mathbf{X} \rightarrow \mathbf{X}$  mit den jeweiligen Kontraktions-Faktoren  $s_n$  für  $n = 1, 2, \dots, N$ . Dann ist der Kontraktions-Faktor des IFS  $s = \max \{s_n : n = 1, 2, \dots, N\}$ . Das IFS wird beschrieben mit  $\{\mathbf{X}; w_n, n = 1, 2, \dots, N\}$ . [Bar95, S 92]

Ein solches IFS beschreibt dann eine Kontraktion auf dem metrischen Raum  $(\mathcal{H}(\mathbf{X}), h)$ , deren Fixpunkt auch eine größere Teilmenge von  $\mathbf{X}$  sein kann, in folgender Weise:

**Satz 3.** Es sei  $\{\mathbf{X}; w_n, n = 1, 2, \dots, N\}$  ein Iteriertes Funktionensystem mit Kontraktions-Faktor  $s$ . Dann ist die Transformation  $W : \mathcal{H}(\mathbf{X}) \rightarrow \mathcal{H}(\mathbf{X})$ , die durch

$$W(B) = \bigcup_{n=1}^N w_n(B)$$

definiert ist, für alle  $B \in \mathcal{H}(\mathbf{X})$  eine Kontraktion auf einem vollständigen metrischen Raum  $(\mathcal{H}(\mathbf{X}), h(d))$  mit Kontraktions-Faktor  $s$ . Das bedeutet

$$h(W(B), W(C)) \leq s * h(B, C)$$

für alle  $B, C \in \mathcal{H}(\mathbf{X})$ . Der einzige Fixpunkt  $A \in \mathcal{H}(\mathbf{X})$  gehorcht

$$A = W(A) = \bigcup_{n=1}^N w_n(A)$$

und wird für alle  $B \in \mathcal{H}(\mathbf{X})$  durch  $A = \lim_{n \rightarrow \infty} W^{\circ n}(B)$  gegeben. [Bar95, S 92]

Um besser mit dem Fixpunkt des IFS arbeiten zu können, erhält dieser einen eigenen Namen.

**Definition 9.** Der Fixpunkt  $A \in \mathcal{H}(\mathbf{X})$  aus Satz 3 heißt der Attraktor des IFS. [Bar95, S 92]

Das Bild eines solchen Attraktors ist somit die Darstellung eines Fraktals, das durch das IFS beschrieben wird. In dieser Arbeit wird im Weiteren lediglich die Erzeugung von Fraktalen anhand von IFS betrachtet. Es existieren allerdings Fraktale, wie zum Beispiel die Mandelbrot-Menge, die nicht durch iterierte Funktionensysteme erzeugt werden.

## 2.2 Der deterministische Algorithmus

Es gibt verschiedene Algorithmen, um den Attraktor eines IFS zu approximieren, zum Beispiel der deterministische oder der Zufalls-Iterations-Algorithmus. Im Folgenden wird der deterministische Algorithmus zur Berechnung von Fraktalen vorgestellt, der auch für die Berechnung im vorliegenden Programm verwendet wird. Dieser ist wie folgt beschrieben.

Sei  $\{\mathbf{X}; w_1, w_2, \dots, w_N\}$  ein IFS. Dann wird eine kompakte, nicht leere Anfangsmenge  $A_0 \subset \mathbf{X}$  gewählt und sukzessiv  $A_n = W^{\circ n}(A)$  gemäß

$$A_{n+1} = \bigcup_{j=1}^N w_j(A_n) \text{ für } n = 1, 2, \dots$$

berechnet. [Bar95, S 98]

Nach Satz 3 konvergiert die Folge  $\{A_n : n = 0, 1, 2, 3, \dots\} \in \mathcal{H}(\mathbf{X})$  gegen den Attraktor des IFS. Das bedeutet, dass man das durch das IFS beschriebene Fraktal beliebig genau approximieren kann, indem man die Anzahl an Iterationen erhöht. Auf diese Weise erzeugt das Programm die Bilder der Fraktale.

### 3 Eigenschaften von Iterierten Funktionensystemen und Anforderungen an deren Transformationen

Im Folgenden werden Eigenschaften von Iterierten Funktionensystemen vorgestellt, um die Berechnung von Fraktalen konfigurieren und ungültige Eingaben erkennen zu können. Da im Rahmen der Anwendung nur die Räume  $\mathbb{R}^2$  und  $\mathbb{R}^3$  im Zusammenhang mit der euklidischen Metrik verwendet werden und die Abbildungen der IFS lineare Abbildungen sind, lassen sich einige Sätze präzisieren und genaue Anforderungen an die Transformationen der IFS stellen. Dabei wird im Folgenden ohne Beschränkung der Allgemeinheit der  $\mathbb{R}^2$  behandelt.

#### 3.1 Skalierung von Fraktalen

Um ein Fraktal um einen Faktor  $s$  zu vergrößern oder zu verkleinern, müssen die konstanten Anteile  $t$  aus den Transformationsgleichungen  $A \cdot x + t$  mit diesem Faktor multipliziert werden.

**Satz 4.** Gegeben ein IFS  $\{\mathbb{R}^2; w_n, n = 1, 2, \dots, N\}$  mit Attraktor  $A$ , wobei  $w_n(x) = A_n \cdot x + t_n$ . Sei  $w'_n(x) = A_n \cdot x + s \cdot t_n, n = 1, 2, \dots, N$ . Dann ist der Attraktor des IFS  $\{\mathbb{R}^2; w'_n, n = 1, 2, \dots, N\}$  definiert durch

$$B = s \cdot A = \{s \cdot x \mid x \in A\}.$$

*Beweis.* Seien  $W(x) = \bigcup_{i=1}^N w_i(x)$  und  $W'(x) = \bigcup_{i=1}^N w'_i(x)$ . Dann gilt:

$$\begin{aligned} W'(B) &= W'(s \cdot A) = \\ &\bigcup_{x \in A} \bigcup_{i=1}^N w'_i(s \cdot x) = \\ &\bigcup_{x \in A} \bigcup_{i=1}^N (A_i \cdot s \cdot x + s \cdot t_i) = \\ &\bigcup_{x \in A} \bigcup_{i=1}^N (s \cdot (A_i \cdot x + t_i)) = \\ &\bigcup_{x \in A} \bigcup_{i=1}^N (s \cdot w_i(x)) = \\ &s \cdot \bigcup_{x \in A} \bigcup_{i=1}^N w'_i(x) = \\ &s \cdot A = B \end{aligned}$$

Demnach ist  $B$  der Attraktor des neuen IFS.

□

### 3.2 Bestimmung einer geeigneten Detailtiefe

Es ist sinnvoll, den Algorithmus so zu gestalten, dass das Ergebnis unabhängig von der Skalierung des IFS ist. Gegeben ein IFS  $\{\mathbb{R}^2; w_n, n = 1, 2, \dots, N\}$  mit Attraktor  $A$ . Dann kann

$$\pi \cdot (\max\{\|x_n\| : n = 1, 2, \dots, N\})^2$$

als ungefähre Schätzung für die Ausdehnung von  $A$  benutzt werden, wobei  $x_n$  jeweils dem Fixpunkt der Kontraktion  $w_n$  entspricht. Dabei wird der Abstand des entferntesten Fixpunktes zum Ursprung als Radius für einen Kreis gewählt.

Mithilfe dieser Abschätzung lässt sich für den Algorithmus eine Detail-Tiefe angeben, für die bei IFS, die sich nur in Bezug auf die Skalierung unterscheiden, Attraktoren entstehen, die sich ebenfalls lediglich bezüglich der Skalierung unterscheiden.

### 3.3 Selbstähnlichkeit

Eine Eigenschaft, die Fraktale, die durch ein IFS erzeugt werden, im Besonderen auszeichnet, ist deren perfekte Selbstähnlichkeit. Dabei bedeutet der Begriff Selbstähnlichkeit, dass beim Vergrößern von Teilen eines Objekts Strukturen entstehen, die dem ursprünglichen Objekt ähneln. Ist die Selbstähnlichkeit perfekt, dann existieren Teilstrukturen, die bis auf eine Skalierung dem ursprünglichen Objekt entsprechen. Diese Eigenschaft folgt direkt aus der Eigenschaft des Attraktors  $A$  für ein IFS  $\{\mathbf{X}; w_n, n = 1, 2, \dots, N\}$ :

$$A = \bigcup_{n=1}^N w_n(A)$$

Diese Gleichung zeigt, dass der Attraktor sich aus  $N$  Kopien seiner selbst zusammensetzt, die lediglich durch eine lineare Abbildung modifiziert werden und für die wiederum das Gleiche gilt. Daher kann man in ein IFS-Fraktal unendlich weit "hineinzoomen" und erhält immer wieder dieselben Strukturen.

### 3.4 Anforderungen an die Matrizen der Transformationen

Im Folgenden werden Bedingungen an die Matrizen eines IFS beschrieben, damit diese konvergieren und ein Fraktal erzeugen.

Um die folgenden Beweise übersichtlicher zu gestalten, wird eine Kurzschreibweise eingeführt.

**Definition 10.** Für eine Matrix  $M \in \mathbb{R}^{2x2}$  und ein Fraktal  $F \in \mathcal{H}(\mathbb{R}^2)$  ist die Multiplikation wie folgt definiert:

$$M \cdot F = \{M \cdot x : \forall x \in F\}$$

### 3.4.1 Kontraktionen im $\mathbb{R}^2$

Da es sich bei den Transformationen des IFS, die im Programm verwendet werden, ausschließlich um Kontraktionen der Form  $f : \mathbb{R}^2 \rightarrow \mathbb{R}^2 : x \mapsto A \cdot x + t$  für  $A \in \mathbb{R}^{2x2}$  und  $t \in \mathbb{R}^2$  handelt, muss der einzige Fixpunkt  $x_0$  der Gleichung

$$x_0 = A \cdot x_0 + t$$

genügen. Das lässt sich folgendermaßen umformen:

$$x_0 = A \cdot x_0 + t \Leftrightarrow$$

$$E_2 \cdot x_0 = A \cdot x_0 + t \Leftrightarrow$$

$$(E_2 - A) \cdot x_0 = t \Leftrightarrow *$$

$$x_0 = (E_2 - A)^{-1} \cdot t$$

Damit der Schritt (\*) gemacht werden und ein eindeutiger Fixpunkt existieren kann, muss  $(E_2 - A)$  invertierbar sein und damit der Gleichung

$$\det(E_2 - A) \neq 0$$

genügen. Das bedeutet, dass ein IFS nur dann ein Fraktal erzeugen kann, wenn jede Transformation diese Bedingung erfüllt.

### 3.4.2 Flächeninhalt von Fraktalen

Eine weitere Anforderung an gültige IFS lässt sich aus den Eigenschaften des Flächeninhalts von Fraktalen ableiten.

Im Folgenden sei die Funktion  $a : \mathcal{H}(\mathbb{R}^2) \rightarrow \mathbb{R}$  so definiert, dass sie ein Fraktal auf dessen Flächeninhalt abbildet. Dabei gilt Folgendes für die Fläche der Vereinigung disjunkter Teilmengen:

**Satz 5.** Seien  $M, N \subset \mathbb{R}^2$ . Dann gilt  $a(M \cup N) = a(M) + a(N) - a(M \cap N)$ .

Falls für ein IFS ein Fraktal existiert, das durch dieses erzeugt werden kann, ist dessen Flächeninhalt endlich. Das bedeutet, dass Folgendes gelten muss:

**Satz 6.**  $\forall x \in \mathcal{H}(\mathbb{R}^2) \exists k \in \mathbb{N} : a(x) \geq a(w_n^{\circ k}(x))$

für  $n = 1, 2, \dots, N$ .

Diese Bedingung ist gegeben, wenn gilt

$$\forall (A, t) \in IFS : |\det(A)| < 1$$

Für den Beweis dieses Satzes werden einige Hilfssätze benötigt.

**Hilfssatz 2.** Sei  $D$  ein beliebiges Dreieck mit den Eckpunkten  $A, B, C \in \mathbb{R}^2$ . Dann existiert eine den Flächeninhalt des Dreiecks beschreibende Matrix, so dass gilt:

$$a(D) = \frac{1}{2} \cdot \left| \det \begin{pmatrix} a_x & b_x \\ a_y & b_y \end{pmatrix} \right|$$

wobei  $\overrightarrow{AB} = \begin{pmatrix} a_x \\ a_y \end{pmatrix}$  und  $\overrightarrow{AC} = \begin{pmatrix} b_x \\ b_y \end{pmatrix}$ .

**Hilfssatz 3.** Für eine beliebige Matrix  $M \in \mathbb{R}^{2x2}$  und eine Matrix  $d \in \mathbb{R}^{2x2}$ , die die Fläche eines Dreiecks gemäß Satz 2 beschreibt, gilt

$$a(M \cdot d) = |\det(M)| \cdot a(d).$$

*Beweis.* Da sich die Matrix aus Differenzen der Dreieckspunkte ergibt und die Matrixmultiplikation eine lineare Abbildung auf den Dreieckspunkten darstellt, kann diese Operation direkt auf der Matrix ausgeführt werden.

$$\begin{aligned} a(M \cdot d) &= \\ \frac{1}{2}|\det(M \cdot d)| &= \\ \frac{1}{2}|\det(M) \cdot \det(d)| &= \\ |\det(M)| \cdot \frac{1}{2}|\det(d)| &= \\ |\det(M)| \cdot a(d) \end{aligned}$$

□

**Hilfssatz 4.** Für ein Fraktal  $F \in \mathcal{H}(\mathbb{R}^2)$  existiert eine Cauchy-Folge  $D_i$  von Mengen von Dreiecken mit  $\lim_{i \rightarrow \infty} D_i = F$ , wobei gleichzeitig gilt  $a(m \cap n) = 0 : \forall i \in \mathbb{N} : \forall m, n \in D_i, m \neq n$

Dieser Satz kann bewiesen werden, indem man zeigt, dass die zusammenhängenden Teilmengen des Fraktales jeweils durch Polygone approximiert werden und somit trianguliert werden können. Die Vereinigung dieser Dreiecke erfüllt dann genau die Eigenschaften des Hilfssatzes. Der Beweis wird allerdings nicht durchgeführt.

**Hilfssatz 5.** Es sei  $D$  eine Dreiecksmenge von  $N$  Dreiecken aus der Cauchy-Folge aus Hilfssatz 4. Sei  $D = d_i : i = 1, 2, \dots, N$ , dann gilt:

$$a(D) = \sum_1^N a(d_i)$$

*Beweis.* Diese Aussage lässt sich per Induktion beweisen.

*Induktions-Basis:* Für  $N = 1$  gilt

$$a(D) = a\left(\bigcup_{i=1}^N d_i\right) = a(d_i) = \sum_1^N a(d_i)$$

*Induktions-Schritt:*  $N - 1 \rightarrow N$

Aufgrund der Eigenschaften der Menge  $D$  und der Induktionsvoraussetzung (\*) gilt:

$$a\left(\left(\bigcup_{i=1}^{N-1} d_i\right) \cap d_N\right) \leq$$

$$a\left(\bigcup_{i=1}^{N-1} (d_i \cap d_N)\right) = *$$

$$\sum_{i=1}^{N-1} a(d_i \cap d_N) =$$

$$\sum_{i=1}^{N-1} 0 = 0$$

$$\Rightarrow a\left(\left(\bigcup_{i=1}^{N-1} d_i\right) \cap d_N\right) = 0$$

Das bedeutet dann:

$$\begin{aligned}
 a(D) &= a\left(\bigcup_{i=1}^N d_i\right) = \\
 a\left(\bigcup_{i=1}^{N-1} d_i \cup d_N\right) &= \\
 a\left(\bigcup_{i=1}^{N-1} d_i\right) + a(d_N) - a\left(\left(\bigcup_{i=1}^{N-1} d_i\right) \cap d_N\right) &= \\
 a\left(\bigcup_{i=1}^{N-1} d_i\right) + a(d_N) &= * \\
 \sum_{i=1}^{N-1} a(d_i) + a(d_N) &= \\
 \sum_{i=1}^N a(d_i)
 \end{aligned}$$

□

Damit lässt sich dann der Beweis für den Satz 6 führen.

*Beweis.* Gegeben ein Fraktal  $F \in \mathcal{H}(\mathbb{R}^2)$  und eine Transformation auf  $\mathbb{R}^2$  beschrieben durch  $(A, t)$ . Zusätzlich sei  $D_i$  eine Cauchy-Folge für  $F$ , wie in Satz 4 definiert, wobei die Elemente der einzelnen Mengen Matrizen gemäß Satz 2 sind. Dann gilt

$$\begin{aligned}
 a(M \cdot F + t) &= \\
 a(M \cdot F) &= \\
 a\left(M \cdot \lim_{i \rightarrow \infty} D_i\right) &= \\
 a\left(M \cdot \lim_{i \rightarrow \infty} \bigcup_{d \in D_i} d\right) &= \\
 \lim_{i \rightarrow \infty} a\left(M \cdot \bigcup_{d \in D_i} d\right) &= \\
 \lim_{i \rightarrow \infty} \sum_{d \in D_i} a(M \cdot d) &= \\
 \lim_{i \rightarrow \infty} \sum_{d \in D_i} (|\det(M)| \cdot a(d)) &= \\
 \lim_{i \rightarrow \infty} \left( |\det(M)| \cdot \sum_{d \in D_i} a(d) \right) &= \\
 |\det(M)| \cdot \lim_{i \rightarrow \infty} a\left(\bigcup_{d \in D_i} d\right) &= \\
 |\det(M)| \cdot a\left(\lim_{i \rightarrow \infty} D_i\right) &= \\
 |\det(M)| \cdot a(F).
 \end{aligned}$$

Daraus folgt: Damit ein wie in Satz 6 gefordertes  $k$  gefunden werden kann, müssen die Determinanten der Matrizen der Transformationen betragsmäßig kleiner oder gleich 1 sein. Dann kann  $k$  immer als 1 gewählt werden.

Der Spezialfall, dass die Determinante gleich 1 ist, muss ebenfalls ausgeschlossen werden, da es sonst möglich ist, mit einem Eingangsfraktal, dessen Flächeninhalt größer ist als der des Attraktors, ein anderes Fraktal zu erzeugen. Dies steht im Widerspruch zu den Eigenschaften des Attraktors.

□

## 4 Programm zur Darstellung von Fraktalen

Das Programm verwendet den Deterministischen Algorithmus aus 2.2 zur Berechnung von Fraktalen. Es überprüft die eingegebenen Matrizen vor der Berechnung auf Gültigkeit gemäß den Bedingungen, die in Kapitel 3 vorgestellt wurden.

### 4.1 Datenstrukturen

Im Rahmen des Programms wurden Klassen für Vektoren und Matrizen angelegt, die alle notwendigen Operationen unterstützen. Außerdem wurde eine Klasse für Transformationen angelegt, die sich aus einem Vektor und einer Matrix zusammensetzt und die dadurch beschriebene Operation ausführen kann. Für die Darstellung der Elemente der Eingabe- und Ergebnismengen wurde eine Polygon-Klasse angelegt. Diese besteht aus beliebig vielen Ecken und verbindet jede mögliche Kombination aus Ecken des Polygons zu Vertices. Aus Performance-Gründen ist es daher sinnvoll entweder drei Ecken für zweidimensionale Eingaben und vier Ecken für dreidimensionale Eingaben zu setzen.

### 4.2 Implementierung des Algorithmus

Der Algorithmus überprüft vor dem Start alle Transformationen des IFS auf die Voraussetzungen, die in Abschnitt 3.4 vorgestellt wurden. Erfüllt mindestens eine dieser Transformationen nicht die Bedingungen wird die Berechnung mit einer entsprechenden Meldung abgebrochen.

Der Algorithmus implementiert einen Rechenschritt, indem er auf jedes Element der Eingangsmenge jede Transformation des IFS anwendet und somit die Ausgangsmenge bildet. Das bedeutet, dass die Größe der aktuellen Menge exponentiell wächst. Da der Deterministische Algorithmus keinen festgelegten Endpunkt hat, sondern das Ergebnis immer genauer approximiert, muss ein Weg gefunden werden, zu bestimmen, wie lange der Algorithmus rechnen soll, so dass das Ergebnis zum einen detailliert genug ist, zum anderen aber die Berechnung nicht zu lange dauert. Zu diesem Zweck wird dem Algorithmus eine Detail-Tiefe übergeben. Dann wird nach dem in Absatz 3.2 vorgestellten Verfahren der Flächeninhalt des Fraktales abgeschätzt. Damit wird  $\frac{\text{geschätzte Fläche}}{\text{Detail-Tiefe}}$  als Schranke für den Flächeninhalt der einzelnen Elemente bestimmt. Das bedeutet, dass nach jeder Iteration alle Elemente der Ausgangsmenge, deren Flächeninhalt kleiner ist als die Schranke, in die Ergebnismenge übernommen werden, während die restlichen Elemente als Eingangsmenge für den nächsten Iterationsschritt verwendet werden. Das wird so lange wiederholt, bis die Eingangsmenge leer ist. Die Forderung, dass die Determinanten der Matrizen aller Transformationen des IFS betragsmäßig kleiner als 1 sind, garantiert, dass der Algorithmus terminiert.

## 4.3 Graphische Darstellung

Für die graphische Oberfläche wird Qt verwendet. Die Anwendung besteht aus einem einzelnen Fenster. Dieses beinhaltet sowohl eine navigierbare 3D-Darstellung des aktuellen Fraktals als auch einen Eingabebereich zur Parametrisierung und Erstellung neuer Fraktale.

### 4.3.1 Die 3D-Darstellung des Fraktals

Dieses Teilfenster der Anwendung zeigt ein mit OpenGL gerendertes Bild des Fraktals. Es ist möglich, mithilfe der Maus das Fraktal zu drehen und die Kamera zu bewegen, um die Details des Fraktals untersuchen zu können. Außerdem befinden sich unterhalb des Bildes drei Checkboxen. Die eine ermöglicht es, das dargestellte Objekt kontinuierlich zu drehen. Die zweite aktiviert die farbige Darstellung des Fraktals. Dabei wird für jede Teilfläche des Fraktals eine Farbe abhängig von deren Normalvektor gewählt. Die dritte Checkbox aktiviert ein Licht, das beliebig positioniert werden kann. Diese Optionen sind vor allem nützlich, um dreidimensionale Objekte besser darstellen zu können.

### 4.3.2 Der Eingabebereich

In diesem Teilfenster befinden sich sowohl Buttons zum Starten/Abbrechen der Berechnung eines Fraktals als auch Eingabefelder für die Transformationen des IFS und für die Eingangsmenge der Berechnung. Dabei sind alle Matrizen und Vektoren aus dem  $\mathbb{R}^3$ , aber es können auch zweidimensionale Fraktale erzeugt werden, indem die z-Werte der Eingabefelder alle auf 0 gesetzt werden. Die Eingabemenge ist ein einzelnes Polygon, das beliebig viele Ecken beinhalten kann. Es existieren außerdem einige Buttons zur Eingabe vordefinierter IFS in das Programm. Diese IFS werden im Folgenden vorgestellt.

## 4.4 Beispiele von erzeugten Fraktalen

Es existieren einige prominente Fraktale, die häufig als Beispiele verwendet werden. Diese werden im Folgenden mithilfe des Programms erzeugt. Dabei werden die Parameter der jeweiligen IFS zur besseren Übersicht in einer Tabelle dargestellt, wobei im Zweidimensionalen jede Zeile die Parameter einer Transformation der Form  $w_i(x) = \begin{pmatrix} a_i & b_i \\ c_i & d_i \end{pmatrix} x + \begin{pmatrix} e_i \\ f_i \end{pmatrix}$  beschreibt, während im Dreidimensionalen Transformationen der Form  $w_i(x) = \begin{pmatrix} a_i & b_i & c_i \\ d_i & e_i & f_i \\ g_i & h_i & i_i \end{pmatrix} x + \begin{pmatrix} j_i \\ k_i \\ l_i \end{pmatrix}$  beschrieben werden.

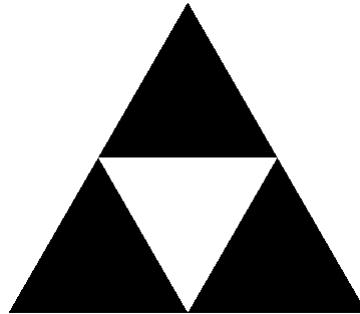
#### 4.4.1 Das Sierpinski-Dreieck im zwei- und dreidimensionalen Raum

Wohl das bekannteste Fraktal ist das Sierpinski-Dreieck im  $\mathbb{R}^2$ . Dieses Fraktal besitzt eine sehr einfache Struktur. Dabei wird aus einem gegebenen Dreieck das ihm einbeschriebene, halb so große und gespiegelte Dreieck entfernt, so dass drei kleinere, gleich große Dreiecke entstehen (siehe Abb. 1). Ein mögliches IFS zur Erzeugung eines solchen Dreiecks ist durch Tabelle 1 beschrieben.

Tabelle 1: Ein IFS für ein 2D Sierpinski-Dreieck

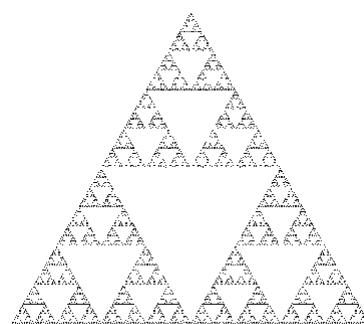
w	a	b	c	d	e	f
1	0.5	0	0	0.5	1	1
2	0.5	0	0	0.5	1	1
3	0.5	0	0	0.5	1	1

Abbildung 1: Ein Iterationsschritt zur Erzeugung eines Sierpinski-Dreiecks



Diese Prozedur wird dann für jedes der neuen Dreiecke wiederholt, so dass immer mehr, immer kleinere Dreiecke entstehen, bis die Dreiecke die Minimalgröße erreicht haben. Dabei entsteht bei einer Detail-Tiefe von 1000 das Bild aus Abbildung 2.

Abbildung 2: Ein 2D Sierpinski-Dreieck



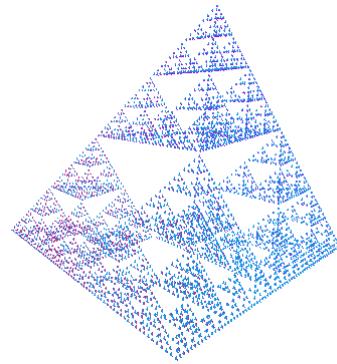
Diese Struktur kann auch auf den dreidimensionalen Raum übertragen werden. Dabei wird aus einem Tetraeder ein halb so großes Tetraeder entfernt und es entstehen vier halb so große neue Tetraeder. Das zugehörige IFS wird in Tabelle 2 dargestellt.

Tabelle 2: Ein IFS für ein Sierpinski-Tetraeder

w	a	b	c	d	e	f	g	h	i	j	k	l
1	0.5	0	0	0	0.5	0	0	0	0.5	0	0	0
2	0.5	0	0	0	0.5	0	0	0	0.5	5	0	0
3	0.5	0	0	0	0.5	0	0	0	0.5	2.5	4.33	0
4	0.5	0	0	0	0.5	0	0	0	0.5	2.5	2.17	3.5

Im Bild für das Sierpinski-Tetraeder (Abb. 4) wurden zusätzlich Farben verwendet, um den dreidimensionalen Aspekt möglichst deutlich hervorzuheben.

Abbildung 3: Ein Sierpinski-Tetraeder



#### 4.4.2 Der Menger-Schwamm

Ein dreidimensionales Fraktal, das eine große Ähnlichkeit zum Sierpinski-Tetraeder aufweist, ist der Menger-Schwamm. Dabei wird ein Würfel in 27 (=3x3x3) gleich große Teilwürfel zerlegt und dann werden sowohl der insgesamt mittlere als auch die mittleren Würfel der Flächen des ursprünglichen Würfels entfernt.

Abbildung 4: Ein Sierpinski-Quader

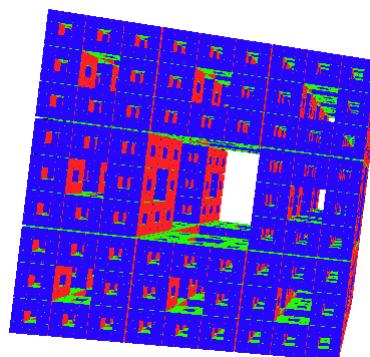


Tabelle 3: Ein IFS für den Menger-Schwamm

w	a	b	c	d	e	f	g	h	i	j	k	l
1	0.33	0	0	0	0.33	0	0	0	0.33	0	0	0
2	0.33	0	0	0	0.33	0	0	0	0.33	10	0	0
3	0.33	0	0	0	0.33	0	0	0	0.33	20	0	0
4	0.33	0	0	0	0.33	0	0	0	0.33	20	10	0
1	0.33	0	0	0	0.33	0	0	0	0.33	20	20	0
2	0.33	0	0	0	0.33	0	0	0	0.33	10	20	0
3	0.33	0	0	0	0.33	0	0	0	0.33	0	20	0
4	0.33	0	0	0	0.33	0	0	0	0.33	0	10	0
1	0.33	0	0	0	0.33	0	0	0	0.33	0	0	0
2	0.33	0	0	0	0.33	0	0	0	0.33	10	0	20
3	0.33	0	0	0	0.33	0	0	0	0.33	20	0	20
4	0.33	0	0	0	0.33	0	0	0	0.33	20	10	20
1	0.33	0	0	0	0.33	0	0	0	0.33	20	20	20
2	0.33	0	0	0	0.33	0	0	0	0.33	10	20	20
3	0.33	0	0	0	0.33	0	0	0	0.33	0	20	20
4	0.33	0	0	0	0.33	0	0	0	0.33	0	10	20
1	0.33	0	0	0	0.33	0	0	0	0.33	0	0	10
2	0.33	0	0	0	0.33	0	0	0	0.33	20	0	10
3	0.33	0	0	0	0.33	0	0	0	0.33	0	20	10
4	0.33	0	0	0	0.33	0	0	0	0.33	20	20	10

#### 4.4.3 Der Barnsley-Farn

Der Autor des Buches, das als Grundlage für diese Arbeit gedient hat, Michael F. Barnsley fand ein IFS mit einem beeindruckenden Attraktor. Dessen Parameter, dargestellt in Tabelle 4, sind dabei weitaus weniger symmetrisch als die für das Sierpinski-Dreieck.

Tabelle 4: Ein IFS für den Barnsley-Farn

w	a	b	c	d	e	f
1	0	0	0	0.16	0	0
2	0.85	0.04	-0.4	0.85	0	1.6
3	0.2	-0.26	0.23	0.22	0	1.6
4	-0.15	0.28	0.26	0.24	0	0.44

Eine Besonderheit des Barnsley-Farns ist, dass dieser eine Art "Ursprung" besitzt. Die erste Transformation  $w_1$  des IFS bildet alles auf Linien mit dem Ursprung  $(0, 0)$  ab. Von dort aus "entwickelt" sich der Farn dann. In Abbildung 5 ist der Farn dargestellt, wobei der Ursprung besonders gekennzeichnet wurde.

Abbildung 5: Ein Barnsley-Farn



## 5 Fazit

In dieser Arbeit wurde eine Möglichkeit vorgestellt, aus einer einfachen Darstellung einiger Matrizen und Vektoren ein kompliziertes Bild zu erzeugen. Wenngleich dieser Vorgang tatsächlich wie eine äußerst effiziente "Kompression" von geometrischen Informationen erscheint, ist bis heute kein effizienter Weg gefunden worden, aus einem gegebenen Bild ein Iteriertes Funktionensystem zu erzeugen. Das hat zur Folge, dass die einzige Möglichkeit, IFS für Fraktale zu finden, darin besteht, das Funktionensystem zu raten und das aus diesem entstandene Bild mit dem gewünschten Fraktal zu vergleichen. Solange auf diesem Gebiet keine Fortschritte gemacht werden, bleibt die fraktale Geometrie vorerst in der Tat hauptsächlich eine Möglichkeit, "schöne Bilder" zu erzeugen, wobei nur sehr wenige praktische Anwendungen dieser Theorie erkennbar sind.

## Literaturverzeichnis

- [Bar95] BARNSLEY, Michael F.: *Fraktale. Theorie und Praxis der Deterministischen Geometrie.* Spektrum, 1995

## **Eidesstattliche Erklärung**

### Eidesstattliche Erklärung zur Bachelorarbeit

Ich versichere, die von mir vorgelegte Arbeit selbstständig verfasst zu haben. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder nicht veröffentlichten Arbeiten anderer entnommen sind, habe ich als entnommen kenntlich gemacht. Sämtliche Quellen und Hilfsmittel, die ich für die Arbeit benutzt habe, sind angegeben. Die Arbeit hat mit gleichem Inhalt bzw. in wesentlichen Teilen noch keiner anderen Prüfungsbehörde vorgelegen.

*Unterschrift :*

*Ort, Datum :*

