

Selbstähnlichkeit als neuartiges Prinzip zur Quellencodierung

Bernd Hürtgen und Frank Müller, Institut für Elektrische Nachrichtentechnik
Rheinisch-Westfälische Technische Hochschule Aachen, 52056 Aachen
Tel.: (0241) 807683, Fax: (0241) 8888196

Kurzfassung: Dieser Beitrag behandelt ein Verfahren zur Quellencodierung von Bildsignalen. Es wird gezeigt, wie selbstähnliche Strukturen in natürlichen Bildvorlagen, die als eine spezielle Form der Redundanz aufgefaßt werden, zur Datenkompression ausgenutzt werden können. Die Grundlagen dieser sogenannten *Fraktalen Codierung* werden eingeführt und anhand von Beispielen aus der Stand- und der Bewegtbildcodierung sowie der Codierung von Farbbildern erläutert.

1 Einleitung

Die bekannten Verfahren zur Datenkompression von Quellensignalen beruhen im wesentlichen auf der Korrelation räumlich bzw. zeitlich eng benachbarter Abtastwerte. Nicht ausgenutzt wird i. allg. die Tatsache, daß sich in Signalen auch statistische Bindungen zwischen nicht benachbarten Teilen finden lassen, die bei Verwendung herkömmlicher Verfahren zur Datenkompression meist ungenutzt bleiben. *Selbstähnlichkeit* kann in diesem Sinne als eine spezielle Form der Redundanz eines Quellensignals betrachtet werden, die auf den Korrelationen voneinander entfernter Signalanteile basiert. Die Abhängigkeiten entspringen dabei der Tatsache, daß viele Teile des Signals in vergrößerter, verkleinerter bzw. geometrisch verzerrter Form an anderer Stelle wieder auftreten. Insbesondere weisen natürliche Signale, wie z. B. Sprache und Bilder, einen hohen Grad an Selbstähnlichkeit auf. Für Verfahren, die diese Art von Bindungen zur Datenkompression nutzen, hat sich in der Literatur der Ausdruck *Fraktale Transformation* eingebürgert.

Ausgehend von der Tatsache, daß deterministische Fraktale, z. B. die bekannte Mandelbrotmenge, eine sehr komplexe Struktur aufweisen, jedoch gleichzeitig durch eine sehr einfache Konstruktionsvorschrift generiert werden können, hat Barnsley vorgeschlagen, das sogenannte *inverse Problem* als Grundlage eines neuen Codiervorgangs zu verwenden [1]. Dieses besteht darin, zu einem gegebenen Signal eine möglichst einfache Konstruktionsvorschrift zu finden, aus der das ursprüngliche Signal zumindest approximativ wieder rekonstruiert werden kann. Da dann das Signal oder dessen Approximation durch die Konstruktionsvorschrift vollständig beschrieben ist, wird ein Codiergewinn erzielt, wenn zur Repräsentation der Konstruktionsvorschrift eine geringere Datenmenge benötigt wird als für das Signal selbst.

Ein besonders breites Anwendungsfeld für die fraktale Transformation findet sich im Bereich der Bildcodierung. Eine erste praktische Implementierung zur Bildcodierung von Grauwertbildern wurde von Jacquin veröffentlicht [2]. Aufbauend auf diesem Verfahren haben eine Reihe von Autoren Modifikationen und Verbesserungen vorgeschlagen, welche die Fraktale Transformation als ein interessantes Verfahren für die Bildcodierung erscheinen lassen, insbesondere wenn große bis sehr große Kompressionsfaktoren erwünscht sind.

Dieser Artikel führt im folgenden Abschnitt 2 die Grundlagen der Fraktalen Transformation ein. Es wird gezeigt, daß diese als eine spezielle Form der Vektorquantisierung mit quel-

lenabhängigem Codebuch aufgefaßt werden kann. Der Abschnitt 3 betrachtet ausführlich die drei Anwendungsfälle Standbildcodierung, Bewegtbildcodierung und Farbbildcodierung. Der Artikel schließt mit einer Zusammenfassung und einigen abschließenden Bemerkungen.

2 Grundlagen

2.1 Norm, Metrik und Eigenwerte

Das aus n Abtastwerten bestehende Signal $\mathbf{x} = (x_1, x_2, \dots, x_n)^T \in \mathbb{R}^n$ wird als Punkt im n -dimensionalen euklidischen Vektorraum aufgefaßt. Die "Länge" dieses Vektors läßt sich durch Angabe einer geeigneten Norm bestimmen. Im euklidischen Vektorraum ist diese durch

$$\|\mathbf{x}\| := \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle} = \sqrt{\sum_{i=1}^n |x_i|^2} \quad (1)$$

definiert, wobei $\langle \cdot \rangle$ das Skalarprodukt darstellt. Durch diese Definition wird \mathbb{R}^n ein normierter Raum und zusätzlich durch Induktion einer Metrik

$$\varrho(\mathbf{x}, \mathbf{y}) := \|\mathbf{x} - \mathbf{y}\|, \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n, \quad (2)$$

die als Fehlermaß dient, ein normierter metrischer Raum. Weiterhin läßt sich eine zur euklidischen Vektornorm konsistente Matrixnorm für quadratische Matrizen durch

$$\|\mathbf{A}\| := \max_{\lambda_i \in \sigma(\mathbf{A}^T \mathbf{A})} \sqrt{|\lambda_i|} \quad (3)$$

in dem Sinne definieren, daß $\|\mathbf{A}\mathbf{x}\| \leq \|\mathbf{A}\| \|\mathbf{x}\|$ gilt. Hierbei bezeichnet λ_i einen Eigenwert und $\sigma(\mathbf{A}^T \mathbf{A})$ das Spektrum, also die Menge aller Eigenwerte der Matrix $\mathbf{A}^T \mathbf{A}$.

2.2 Fraktale Approximation (Codierung)

Zum Zweck der Codierung wird das gesamte Signal in $N_R = n/n_R$ nicht-überlappende Blöcke eingeteilt, wobei n die Anzahl der Abtastwerte des gesamten Signals und n_R die eines einzelnen Blocks darstellt. Für jeden dieser N_R Blöcke wird dann ein Codebucheintrag \mathbf{y}_j aus einem noch zu definierenden Codebuch mit N_D Einträgen gesucht, der nach geeigneter Skalierung mit α_{ij} und Addition eines Offsets $b_i \mathbf{1}$ das Fehlermaß

$$\varrho(\mathbf{x}_i, \hat{\mathbf{x}}_i) = \|\mathbf{x}_i - \hat{\mathbf{x}}_i\| = \|\mathbf{x}_i - (\alpha_{ij} \mathbf{y}_j + b_i \mathbf{1})\| \quad (4)$$

minimiert. Dieser Ersatzwert ist bei gegebenem Codebuch für jeden Block dann vollständig durch den Parametervektor (α_{ij}, b_i, j) bestimmt, wobei j als Codebuchindex für den entsprechenden Block aufgefaßt wird.

Wie bereits einleitend erwähnt, ist das Codebuch und damit auch alle Codebuchvektoren \mathbf{y}_j selbst abhängig vom zu codierenden Signal \mathbf{x} und kann aus diesem mittels einer einfachen Konstruktionsvorschrift, die sowohl dem Coder als auch dem Decoder bekannt ist, generiert werden. Damit ist im Gegensatz zu herkömmlichen Verfahren der Vektorquantisierung bei der Fraktalen Transformation kein aufwendiges Training des Codebuches erforderlich.

Bezeichnet man mit \mathbf{F}_i die Matrix, die für einen Block \mathbf{x}_i den geeignetsten Codebucheintrag $\mathbf{y}_j = \mathbf{F}_i \mathbf{C} \mathbf{x}$ auswählt und mit \mathbf{P}_i die Matrix, die den entsprechenden Ersatzvektor $\hat{\mathbf{x}}_i = \alpha_{ij} \mathbf{y}_j + b_i \mathbf{1}$ an die richtige Stelle des approximierten Signals $\hat{\mathbf{x}}$ kopiert, so läßt sich für das gesamte Signal der Vorgang der Approximation oder Codierung wie folgt darstellen:

$$\hat{\mathbf{x}} = \sum_{i=1}^{N_R} \mathbf{P}_i (\alpha_{ij} \mathbf{F}_i \mathbf{C} \mathbf{x} + b_i \mathbf{1}) = \left\{ \sum_{i=1}^{N_R} \mathbf{P}_i \alpha_{ij} \mathbf{F}_i \mathbf{C} \right\} \mathbf{x} + \sum_{i=1}^{N_R} \mathbf{P}_i b_i \mathbf{1} = \mathbf{A} \mathbf{x} + \mathbf{b}. \quad (5)$$

Gleichung (5) beschreibt eine affine Transformation mit einem linearen Teil \mathbf{A} und einem nicht-linearen Offset \mathbf{b} . Die Matrix \mathbf{A} und den Vektor \mathbf{b} bezeichnet man als den *fraktalen Code*, aus dem sich ohne Kenntnis des Codebuches unter bestimmten Bedingungen das ursprüngliche Signal \mathbf{x} wieder rekonstruieren läßt. Ein Codiergewinn kann daher dann erwartet werden, wenn sich der fraktale Code durch eine geringere Datenmenge repräsentieren läßt als das Signal selbst.

2.3 Decodierung

Kann zu einem gegebenen Signal \mathbf{x} ein Parameterpaar (\mathbf{A}, \mathbf{b}) gefunden werden, für das eine fehlerfreie Approximation möglich ist, also falls $\|\mathbf{x} - \hat{\mathbf{x}}\| = 0$, so bezeichnet man mit $\mathbf{x}_f = \mathbf{x} = \hat{\mathbf{x}}$ den *Fixpunkt* der durch

$$\mathbf{x}_f = \mathbf{A} \mathbf{x}_f + \mathbf{b} \quad (6)$$

definierten Operatorgleichung. Der *Banachsche Fixpunktsatz* [3] gibt sowohl eine Bedingung für die Lösbarkeit als auch eine iterative Lösungsvorschrift für diese Operatorgleichung an. Dieser sagt aus, daß der Fixpunkt \mathbf{x}_f die einzige Lösung der Operatorgleichung (6) ist und eindeutig durch das Iterationsverfahren

$$\mathbf{x}_f = \lim_{i \rightarrow \infty} \mathbf{x}_i; \quad \mathbf{x}_i = \mathbf{A} \mathbf{x}_{i-1} + \mathbf{b}; \quad \forall \mathbf{x}_0 \in \mathbb{R}^n \quad (7)$$

für jeden beliebigen Startvektor \mathbf{x}_0 bestimmt werden kann. Voraussetzung für eine eindeutige Lösung ist jedoch die *Kontraktivität* der Abbildung $W : \mathbf{x} \rightarrow \mathbf{A} \mathbf{x} + \mathbf{b}$. Dies ist dann der Fall, wenn eine Konstante $s < 1, s \in \mathbb{R}$ existiert, für die $\varrho(W(\mathbf{x}), W(\mathbf{y})) \leq s \varrho(\mathbf{x}, \mathbf{y}) \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ gilt. Mit der Definition der Metrik (2), der Operatornorm (3) und der Abbildungsvorschrift (5) erhält man damit als hinreichende Bedingung für die Kontraktivität der Abbildung W

$$\|\mathbf{A}\| \leq s < 1. \quad (8)$$

Die Ableitung einer notwendigen und hinreichenden Bedingung findet sich in [4].

Im allgemeinen läßt sich jedoch kein Parameterpaar (\mathbf{A}, \mathbf{b}) finden, so daß das zu codierende Signal \mathbf{x} gleichzeitig auch Fixpunkt der Operatorgleichung (6) ist, d. h. in diesem Fall ist $\|\mathbf{x} - \hat{\mathbf{x}}\| > 0$. Das *Collage Theorem* [1, 5, 6] liefert dann jedoch eine obere Abschätzung für den Fehler bei der Rekonstruktion

$$\|\mathbf{x} - \mathbf{x}_f\| \leq \frac{1}{1 - \|\mathbf{A}\|} \|\mathbf{x} - \hat{\mathbf{x}}\| \quad (9)$$

und damit eine Motivation für die Minimierung des Approximationsfehlers $\|\mathbf{x} - \hat{\mathbf{x}}\|$ bei der Codierung.

Zusammenfassend läßt sich die Fraktale Transformation wie folgt beschreiben: Zu einem gegebenen zu codierenden Signal x wird unter Beibehaltung der Kontraktivitätsbedingung $\|A\| < 1$ blockweise eine Approximation $\hat{x} = Ax + b$ gesucht, die den Approximationsfehler $\|x - \hat{x}\|$ minimiert. Das Parameterpaar (A, b) dient als *fraktaler Code* des zu codierenden Signals. Aus diesem fraktalen Code generiert der Decoder iterativ den Fixpunkt der Operatorgleichung $x_f = Ax_f + b$, der aufgrund der Abschätzung (9) eine Approximation des originalen Signals ist. Da bei der fraktalen Transformation nicht das Signal x selbst, sondern der als Approximation dienende Fixpunkt einer Operatorgleichung codiert wird, nennt man diese Art der Transformation auch *Fixpunkt-* oder *Attraktor-Codierung* [7, 8].

3 Anwendungen

Die Anfänge der Fraktalen Transformation auf dem Gebiet der Bildverarbeitung wurden vor allem durch die Modellierung binärwertiger, natürlich aussehender Bilder geprägt. Besonders bekannte Beispiele sind die von Barnsley erzeugten Farne [1, 9]. Durch die Betrachtung einer lokalen Punktdichte bei binärwertigen Bildern können aus diesen leicht Graustufenbilder generiert werden. Wie einleitend bereits erwähnt, ist die Aufgabe im Rahmen der Bildcodierung jedoch nicht die Generierung eines Bildes aus einem Code, sondern die Erzeugung eines beschreibenden Codes zu einem gegebenen Bild.

Dieser Abschnitt gliedert sich in drei Teile. Im ersten Teil wird ausführlich eine Anwendung der Fraktalen Transformation für die Codierung von natürlichen Grauwertbildern aufgezeigt. Wie im Unterpunkt 3.2 gezeigt, läßt sich unter Verwendung eines hybriden Coderkonzeptes das vorgestellte Verfahren leicht für die Codierung von Bildsequenzen erweitern, wobei die zeitlichen Korrelationen in einer DPCM-Schleife ausgenutzt werden. Ebenso ist eine Erweiterung für die Codierung von Farbbildern möglich. Im Unterpunkt 3.3 wird aufgezeigt, wie die Korrelationen zwischen den einzelnen Farbauszügen eines natürlichen Bildes durch die Fraktale Transformation zur Datenkompression genutzt werden können.

3.1 Standbildcodierung

Der zur Zeit vielversprechendste Ansatz zur fraktalen Codierung von Grauwertbildern wurde von Jacquin vorgestellt [2]. Er basiert im wesentlichen auf der blockweisen Approximation eines Bildes durch andere, affin transformierte Blöcke desselben Bildes. Vereinfacht dargestellt läuft der Codiervorgang dabei wie folgt ab: Zunächst wird das gesamte Bild in quadratische, nicht überlappende sogenannte *Range-Blöcke* eingeteilt. Zu jedem dieser Range-Blöcke wird dann ein anderer Block, der *Domain-Block*, innerhalb desselben Bildes gesucht, der nach einer affinen Transformation mit noch zu bestimmenden Parametern im Sinne der Abbildungsgleichung (5) das vorgegebene Fehlermaß (4) minimiert.

Die affine Transformation ist hierbei nicht beliebig, sondern kann nur aus einer endlichen Anzahl von möglichen Transformationen ausgewählt werden. Ebenfalls ist die Anzahl der Domain-Blöcke begrenzt, damit einerseits der Suchaufwand in Grenzen gehalten und andererseits noch ein Codiergewinn erzielt werden kann. Die Menge der möglichen Domain-Blöcke zusammen mit der Menge der erlaubten affinen Transformationen bildet dann das Codebuch für das gegebene Bild. Da die erlaubten Transformationen ebenso wie die Lage der Domain-Blöcke auch dem Decoder bekannt sind, besteht der fraktale Code für jeden Range-Block lediglich aus dem Domain-Block-Index, einer Kennzeichnung für den Typ der

verwendeten affinen Transformation, dem Skalierungsfaktor α und dem Offset b . Der iterative Decodiervorgang startet mit einem beliebigen Initialbild. Genau wie bei der Approximation während der Codierung werden die durch den fraktalen Code definierten Domain-Blöcke entsprechend ihrer Transformationsvorschrift an die Stelle der entsprechenden Range-Blöcke kopiert. Aufgrund des Banachschen Fixpunktsatzes konvergiert die Folge von Iterationen gegen den Fixpunkt der Abbildungsgleichung (5), der nach Voraussetzung eine Approximation des originalen Bildes ist, so daß

$$\mathbf{x}_i = \mathbf{A}\mathbf{x}_{i-1} + \mathbf{b}; \quad \mathbf{x} \approx \mathbf{x}_f = \lim_{i \rightarrow \infty} \mathbf{x}_i. \quad (10)$$

Eine typische Folge von rekonstruierten Bildern wird in **Bild 1** gezeigt, wobei als Initialbild ein einheitliches Bild mittleren Grauwertes verwendet wurde.



Bild 1 Folge von rekonstruierten Bildern aus dem fraktalen Code nach einer, zwei und fünf Iterationen.

Im Laufe der letzten Jahre wurden eine Reihe von Verbesserungen und Modifikationen vorgestellt, die zum einen das Ziel hatten die Rekonstruktionsqualität zu verbessern und zum anderen den erheblichen Rechenaufwand bei der Codierung zu verringern.

Wie aus (9) ersichtlich, läßt sich die Rekonstruktionsqualität verbessern, indem man den Fehler bei der Approximation $\|\mathbf{x} - \hat{\mathbf{x}}\|$ möglichst klein macht. Dieses kann zum Beispiel durch eine kompliziertere als die uniforme Blocksegmentierung des zu codierenden Signals erreicht werden. Interessante Ansätze hierzu finden sich in [10], wo verschiedene Partitionierungen, z. B. Quadrees, Dreiecke o. ä. verwendet werden. Ebenso kann wie in [11] der nicht-lineare Offset \mathbf{b} als eine Linearkombination von mehreren Basisfunktionen ausgebildet werden, wodurch auf Kosten des Kompressionsfaktors eine Verbesserung der Rekonstruktionsqualität erreicht werden kann.

Um den enormen Rechenaufwand bei der Codierung zu reduzieren, wurden ähnlich wie bei der Vektorquantisierung, eine Codebuchklassifizierung und eine hierarchische Codebuchsuche eingeführt, z. B. [12, 13].

Besonders wichtig sind auch die Untersuchungen bezüglich der Konvergenzeigenschaften der Abbildungsgleichung, da diese eine notwendige Bedingung für das Funktionieren der Fraktalen Transformation ist. Die Forderung nach einem kleinen Rekonstruktionsfehler und die Erfüllung der Kontraktivitätsbedingung (8) sind, von Ausnahmen abgesehen, nämlich kontraproduktiv. Einige grundlegende theoretische Untersuchungen zur Kontraktivität, die auf der Betrachtung des Spektralradius der Transformationsmatrix \mathbf{A} basieren, sind in [6, 4, 14] zu finden.

3.2 Bewegtbildcodierung

Die Verbreitung digitaler Netzwerke innerhalb der letzten Jahre setzt auch effizientere Techniken der Datenkompression voraus, da die Zunahme des zu übertragenden oder zu speichernden Datenvolumens deutlich größer ist als die der verfügbaren Kanal- bzw. Speicherkapazität. Insbesondere bildbasierende Dienste, z. B. Bildtelefonie, haben einen enormen Bedarf an Übertragungsbandbreite. Da die bekannten Verfahren, wie z. B. H.261 basierte, für die Codierung von Bildsequenzen bei sehr niedrigen Datenraten weniger geeignet erscheinen, werden von der "MPEG Ad-Hoc Group on Very-Low Bitrate Audio-Visual Coding" neben modellbasierten Verfahren auch fraktale Verfahren als mögliche Lösung angeführt.

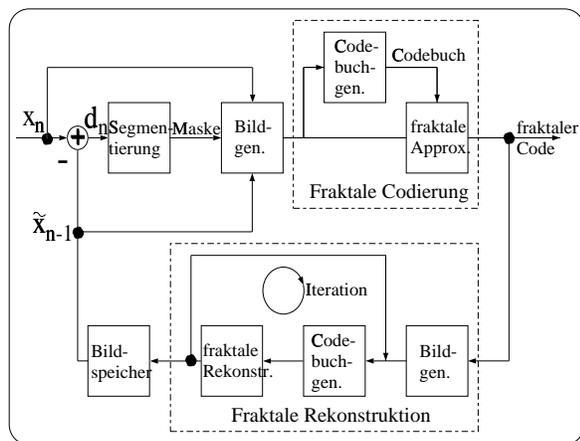


Bild 2 Blockdiagramm des hybriden Coderkonzeptes mit fraktaler Codierung der nicht prädizierbaren Bildanteile.

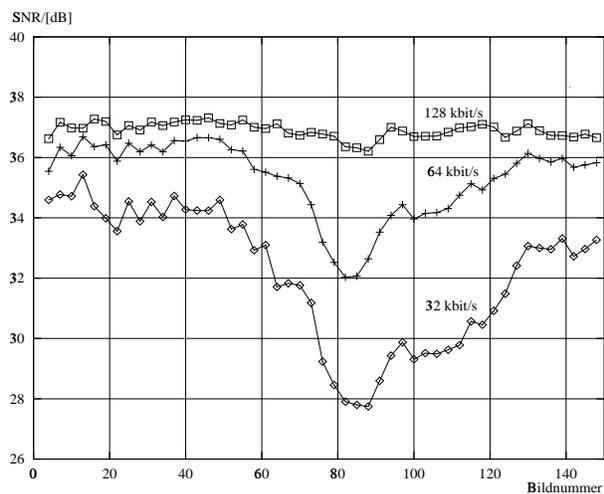


Bild 3 Simulationsergebnisse für die Testsequenz "Miss America" bei verschiedenen Datenraten.

Ein Ansatz, der die Fraktale Transformation auch für die Codierung von Bildsequenzen verwendet, wurde in [15] veröffentlicht. Ausgangspunkt ist das hybride Coderkonzept, bei dem die zeitlichen Korrelationen zwischen zwei aufeinanderfolgenden Bildern durch eine DPCM-Schleife ausgenutzt werden. Zusätzlich kann eine bewegungskompensierte Prädiktion angewendet werden, die eine signifikante Reduktion der Fehlerleistung des Prädiktionsbildes bewirkt. Die Stellen des Bildes, für die die Prädiktion versagt, werden dann mit Hilfe der oben beschriebenen Fraktalen Transformation codiert und zum Empfänger übertragen. **Bild 2** zeigt das Blockdiagramm eines modifizierten Encoders und **Bild 3** die Simulationsergebnisse für eine Standardbildsequenz bei verschiedenen Datenraten.

3.3 Farbbildcodierung

Obwohl die meisten Untersuchungen zur Fraktalen Transformation die Codierung von Grauwertbildern behandelt, ist im Sinne eines allgemein verwendbaren Codierschemas auch die Berücksichtigung der Farbinformation vonnöten [16].

Die Physiologie des menschlichen Auges ist die Basis für die trichromatische Theorie des Farbsehens. Diese besagt, daß die Farbe des beobachteten Lichts ein Gemisch von lediglich drei Grundfarben ist. Gewöhnlich werden die Farben Rot, Grün und Blau als Primärfarben der additiven Farbmischung und Gelb, Magenta und Zyan als Primärfarben der subtraktiven

Farbmischung verwendet. Ein Farbbild läßt sich damit als Komposition von drei verschiedenen Farbebenen auffassen, die aufgrund des physikalischen Bilderzeugungsprozesses starke statistische Bindungen besitzen.

Jede Farbebene unabhängig von allen anderen zu betrachten ist die einfachste Methode Farbbilder zu codieren. Hierdurch wird jedoch eine große Datenmenge für redundante Informationen verschwendet. Besser ist daher eine gemeinsame Codierung aller Farbauszüge, wie sie am Beispiel des RGB-Farbraumes im folgenden beschrieben wird. Zunächst wird die willkürlich wählbare Hauptkomponente, z. B. der Grünauszug, unabhängig von allen anderen codiert, wie in 3.1 gezeigt. Hieraus erhält man den fraktalen Code $(\mathbf{A}, \mathbf{b})_G$ für die Grünkomponente. In einem ersten Schritt wird dieser Code auch den beiden abhängigen Komponenten Rot und Blau zugewiesen, so daß $(\mathbf{A}, \mathbf{b})_{R,B} = (\mathbf{A}, \mathbf{b})_G$. Für einige Blöcke der abhängigen Komponenten ist dies natürlich eine unzureichende Beschreibung, so daß für diese Blöcke eine sukzessive Modifikation vorgenommen wird, bis eine ausreichende Rekonstruktionsqualität für alle Farbebenen erreicht ist. Die fraktalen Codes der abhängigen Komponenten $(\mathbf{A}, \mathbf{b})_R$ und $(\mathbf{A}, \mathbf{b})_B$ sind dann lediglich leichte Modifikationen des Codes der Hauptkomponente $(\mathbf{A}, \mathbf{b})_G$. Durch Einführen sogenannter Übertragungsfunktionen U_R und U_B , lassen sich die Codes der abhängigen Komponenten als Funktion des Codes der Hauptkomponente darstellen, so daß $(\mathbf{A}, \mathbf{b})_R = U_R((\mathbf{A}, \mathbf{b})_G)$; $(\mathbf{A}, \mathbf{b})_B = U_B((\mathbf{A}, \mathbf{b})_G)$. Da die Modifikationen im Code der abhängigen Komponente meist nur gering sind, lassen sich die Übertragungsfunktionen durch eine geringere Datenmenge repräsentieren als die zugehörigen fraktalen Codes. Der fraktale Code für das gesamte Farbbild setzt sich dann zusammen aus dem Code für die Hauptkomponente $(\mathbf{A}, \mathbf{b})_G$ und den beiden Übertragungsfunktionen U_R und U_B .

4 Zusammenfassung

In diesem Artikel wurde ein Codierverfahren vorgestellt, das die Selbstähnlichkeit von natürlichen Signalen zur Codierung ausnutzt. Die Selbstähnlichkeit wurde als eine besondere Form der Redundanz eingeführt, die neben den Korrelationen räumlich und zeitlich benachbarter Abtastwerte auch über größere Distanzen im Signal vorhanden sein kann. Als selbstähnlich oder zumindest teilweise selbstähnlich wurden solche Signale bezeichnet, bei denen Teile des Signals durch einfache affine Transformationen anderer Teile desselben Signals approximiert werden können. Das vorgestellte Codierverfahren kann als Sonderfall der Vektorquantisierung interpretiert werden, bei der das Codebuch nicht fest, sondern abhängig vom zu codierenden Signal ist. Dieses Codebuch muß dem Decoder nicht bekannt sein, da dieser es unter der Voraussetzung einer kontraktiven Abbildung iterativ rekonstruieren kann. Nach Erläuterung der notwendigen mathematischen Grundlagen wurden im dritten Abschnitt drei Anwendungsfälle diskutiert. Die Codierung von Grauwertbildern stellt dabei das breiteste Anwendungsfeld der Fraktalen Transformation dar. Hierauf aufbauend wurde erläutert, wie unter Verwendung des hybriden Coderkonzeptes die Codierung von Bildsequenzen möglich ist. Zum Schluß wurde gezeigt, wie die statistischen Bindungen zwischen den verschiedenen Farbebenen eines natürlichen Farbbildes mit Hilfe der Fraktalen Transformation zur Datenkompression ausgenutzt werden können.

5 Literatur

- [1] M. F. Barnsley, *Fractals Everywhere*. London: Academic Press Inc., 1988.
- [2] A. E. Jacquin, "Fractal image coding based on a theory of iterated contractive image transformations," In *Proceedings SPIE Visual Communications and Image Processing '90*, Bd. 1360, S. 227–239, 1990.
- [3] E. Kreyszig, *Introductory functional analysis with applications*. Robert E. Krieger Publishing Company, 1989.
- [4] B. Hürtgen und S. F. Simon, "On the problem of convergence in fractal coding schemes," In *Proceedings of the IEEE International Conference on Image Processing ICIP'94*, Bd. 3, (Austin, Texas, USA), S. 103–106, November 1994.
- [5] Y. Fisher, E. W. Jacobs, und R. D. Boss, "Fractal image compression using iterated transforms," In *Image and text compression* (J. A. Storer, Hrsg.), Kap. 2, S. 35–61, Kluwer Academic Publishers, 1992.
- [6] L. Lundheim, *Fractal signal modelling for source coding*. Dissertation, Universitetet I Trondheim Norges Tekniske Høgskole, 1992.
- [7] S. Lepsøy, *Attractor image compression - Fast algorithms and comparisons to related techniques*. Dissertation, Universitetet I Trondheim Norges Tekniske Høgskole, 1993.
- [8] G. E. Øien, *L_2 -optimal attractor image coding with fast decoder convergence*. Dissertation, Universitetet I Trondheim Norges Tekniske Høgskole, 1993.
- [9] M. F. Barnsley und L. P. Hurd, *Fractal Image Compression*. New-York: Springer-Verlag, 1993.
- [10] H. O. Peitgen, H. Jürgens, und D. Saupe, *Bausteine des Chaos: Fraktale*. Klett-Cotta/Springer-Verlag, 1992.
- [11] G. E. Øien, S. Lepsøy, und T. A. Ramstad, "An inner product space approach to image coding by contractive transformations," In *Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing ICASSP'91*, Bd. 4, (Toronto, Canada), S. 2773–2776, 1991.
- [12] G. E. Øien, S. Lepsøy, und T. A. Ramstad, "Reducing the complexity of a fractal based image coder," In *Proceedings of the EUSIPCO'92*, S. 1353–1356, 1992.
- [13] B. Hürtgen und C. Stiller, "Fast hierarchical codebook search for fractal coding of still images," In *Proceedings EOS/SPIE Visual Communications and PACS for Medical Applications '93*, Bd. 1977, (Berlin, Germany), S. 397–408, 1993.
- [14] B. Hürtgen und T. Hain, "On the convergence of fractal transforms," In *Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing ICASSP'94*, Bd. 5, S. 561–564, 1994.
- [15] B. Hürtgen und P. Büttgen, "Fractal approach to low rate video coding," In *Proceedings of the International Conference on Visual Communications and Image Processing, SPIE '93*, Bd. 2094, (Cambridge, Massachusetts, USA), S. 120–131, November 1993.
- [16] B. Hürtgen, P. Mols, und S. F. Simon, "Fractal transform coding of color images," In *Proceedings of the International Conference on Visual Communications and Image Processing, SPIE '94*, Bd. 2308, (Chicago, Illinois, USA), S. 1683–1691, September 1994.