Fourierrekonstruktion in der Computer-Tomographie

Diplomarbeit

von

Jörg Schulte

Februar 1994

angefertigt nach einer Aufgabenstellung von Prof. Dr. F. Natterer am Institut für numerische und instrumentelle Mathematik des Fachbereiches 15 der Westfälischen Wilhelms-Universität Münster.

Hiermit erkläre ich an Eides Statt, daß ich diese Arbeit selbständig angefertigt und nur die im Literaturverzeichnis angegebenen Quellen benutzt habe.

Münster, im Februar 1994

Inhaltsverzeichnis

Abbildungsverzeichnis

Einleitung

Ein Spezialfall der Computer-Tomographie besteht in der Rekonstruktion einer Scheibe (gr. Tomos) eines Körpers durch die Messung der Absorption von Röntgenstrahlen, die diesen durchdringen. Ein einfaches mathematisches Modell für diesen Vorgang ist die Berechnung einer Funktion mit endlichem Träger aus einer endlichen Menge ihrer Linienintegrale, wobei deren Anordnung und Zahl, die Abtastgeometrie, die mögliche Rekonstruktionsgüte einschränkt. Es soll hier nur der praktisch wesentliche zweidimensionale Fall betrachtet werden, wobei jedoch die wichtigen mathematischen Sätze in allgemeiner Form bewiesen werden. Die zu besprechenden Verfahren sind auf die parallele, äquidistante Anordnung der Linienintegrale aus der jeweiligen Richtung ausgelegt (Standard-Parallelgeometrie, siehe 1.3), zur Umrechnung von Fächerstrahlgeometrien (engl. fan-beam) siehe [9].
Ein als Projektionssatz oder Fourierscheibentheorem bekanntes Ergebnis liefert folgendes : Die eindimensionale Fouriertransformation einer Parallel–Projektion entspricht der Fouriertransformation des Objekts auf einem Zentralstrahl, der senkrecht zur Projektionsrichtung liegt. Die Diskretisierung und Implementation dieses Theorems wird als Fourierrekonstruktion bezeichnet. Dabei ist die effektive Interpolation der Fourierwerte von einem Polargitter auf ein kartesisches Gitter das eigentliche Problem und Thema dieser Arbeit.
Eine sehr bekannte Technik bezüglich dieses Interpolationsproblems ist die sogenannte bilineare Methode. Es wurde schon häufig experimentell nachgewiesen, daß dieses Verfahren, wie auch ähnliche Lösungsversuche, keine guten Ergebnisse liefert [1, 9]. In dieser Arbeit werden im wesentlichen drei Algorithmen vorgestellt, die sehr viel bessere Rekonstruktionen erzeugen, ohne den bekannten Geschwindigkeitsvorteil der Fourierrekonstruktion gegenüber der gefilterten Rückprojektion [1, 9] aufzugeben. Bei diesem Standardalgorithmus der Computer-Tomographie findet die Interpolation im Gegensatz zur Fourierrekonstruktion im Ortsraum statt. Im letzten Kapitel werden die Fouriermethoden auch hinsichtlich der Bildqualität mit diesem Standardverfahren verglichen.

Kapitel 1

Grundlagen

1.1 Fouriertransformation und FFT

Es soll zuerst auf die *Fourier-Transformation* und ihre Diskretisierung eingegangen werden. Ferner wird der FFT -Algorithmus behandelt, der zur Berechnung der diskreten Fouriertransformation dient.

Definition 1.1 *Betrachte den Vektorraum aller integrierbaren komplexwertigen Funktionen $f : \mathbb{R}^n \to \mathbb{C}$.*
Die durch

$$\hat{f}(\xi) := (2\pi)^{-n/2} \int\limits_{\mathbb{R}^n} f(x)\, e^{-i\,<x,\xi>} dx$$

definierte Funktion $\hat{f} : \mathbb{R}^n \to \mathbb{C}$ heißt die Fourier-Transformierte von f.
Die durch

$$\tilde{f}(\xi) := (2\pi)^{-n/2} \int\limits_{\mathbb{R}^n} f(x)\, e^{i\,<x,\xi>} dx$$

definierte Funktion $\tilde{f} : \mathbb{R}^n \to \mathbb{C}$ heißt die inverse Fourier-Transf. von f.

Die Fouriertransformation ist ein Automorphismus über dem Vektorraum $\mathcal{S}(\mathbb{R}^n)$, dem Schwartzschen Raum, dort gilt $\tilde{\hat{f}}=f$, was die Bezeichnung *inverse Fourier-Transformation* erklärt.

Definition 1.2 *Der Schwartzsche Raum $\mathcal{S}(\mathbb{R}^n)$ ist der Vektorraum aller Funktionen $f \in C^\infty(\mathbb{R}^n)$ mit der Eigenschaft* $\sup_{x\in\mathbb{R}^n} |x^\beta\ D^\alpha f(x)| < \infty$ *für alle $\alpha, \beta \in \mathbb{N}^n$*

$\mathcal{S}(\mathbb{R}^n)$ wird auch intuitiv als Raum der schnell fallenden, unendlich oft differenzierbaren Funktionen bezeichnet. Es gilt $C_c^\infty(\mathbb{R}^n) \subset \mathcal{S}(\mathbb{R}^n)$.

Betrachte nun Fourierreihen periodischer Funktionen in $C^\infty(\mathbb{R}^n)$:
$f \in C^\infty(\mathbb{R}^n)$ sei in allen Argumenten $2a-$periodisch, dann gilt mit gleichmäßiger Konvergenz

$$f(x) = \sum_{k \in Z^n} \hat{f}_k \, e^{i\pi<x,k>/a}$$

mit Fourierkoeffizienten

$$\hat{f}_k = (2a)^{-n} \int\limits_{[-a,a]^n} f(x) \, e^{-i\pi<x,k>/a} dx$$

Denkt man sich die Periode $2a$ ins Unendliche vergrößert, so kann man die analytische Fouriertransformation aus Definition 1.1 als kontinuierliches Analogon dieser Fourierreihen betrachten.
Es ist noch zu bemerken, daß mit dieser Notation auch Fourierreihen in $L_2([-a,a]^n)$ beschrieben werden, dort aber keine gleichmäßige , sondern nur Konvergenz im quadratischen Mittel (im Sinne der L_2-Norm) vorliegt. Die Fouriertransformation auf $L_2(\mathbb{R}^n)$ selbst ist durch den Satz von Plancherel [16] gerechtfertigt. Dort ist $\hat{f}$ nur bis auf Gleichheit fast überall – d.h. bis auf eine Nullmenge – eindeutig bestimmt, im Gegensatz zum Fall $f \in L_1(\mathbb{R}^n)$, in dem $\hat{f}$ eine stetige Funktion ist. Als Abbildung über dem Hilbertraum $L_2(\mathbb{R}^n)$ gesehen, ist die Fouriertransformation ein Automorphismus, normerhaltend und damit stetig.

Es sollen jetzt ohne Beweis die wichtigsten Eigenschaften der Fouriertransformation aufgeführt werden .
Sei $f \in \mathcal{S}(\mathbb{R}^n)$, dann gelten folgende Regeln :

1. Sei $r > 0$ und $f_r(x) := f(rx)$, dann gilt:

   $$\hat{f}_r(\xi) = r^{-n} \, \hat{f}\left(\frac{1}{r}\,\xi\right) \quad (*)$$

2. Für $y \in \mathbb{R}^n$ definiere die Translation von f durch $(\tau_y f)(x) := f(x+y)$. Dann gilt:

   $$(\tau_y f)\hat{}(\xi) = e^{i<\xi,y>} \, \hat{f}(\xi) \quad (*)$$

3. Für $k \in \mathbb{N}^n$ gilt :

   $$(D^k f)\hat{} = i^{|k|} \, \xi^k \, \hat{f}, \quad (x^k f)\hat{} = i^{|k|} \, D^k \hat{f}$$

   mit $|k| := k_1 + \ldots + k_n, \quad \xi^k := \xi_1^{k_1} \cdots \xi_n^{k_n}$.

4. Der Faltungssatz. Sei $g \in \mathcal{S}(\mathbb{R}^n)$, dann gilt:

$$\widehat{(f * g)} = (2\pi)^{n/2}\, \hat{f}\, \hat{g}, \quad \widehat{(fg)} = (2\pi)^{-n/2}\, \hat{f} * \hat{g} \quad (*)$$

   Die Faltung ist definiert durch :

$$(f * g)\,(x) := \int\limits_{\mathrm{R}^n} f(x-y)\, g(y)\, dy$$

5. Für $f, g \in \mathcal{S}(\mathbb{R}^n)$ oder $f, g \in L_2(\mathbb{R}^n)$ gilt die Parsevalsche Beziehung:

$$\int\limits_{\mathrm{R}^n} f\, \hat{g}\, dx = \int\limits_{\mathrm{R}^n} \hat{f}\, g\, dx$$

6. Zur Beziehung mit der inversen Fouriertransformation: $\overline{\hat{f}} = \overline{\tilde{f}}$. (*)

Die Eigenschaften 1-5 lassen sich durch Eigenschaft 6 auf die inverse FT übertragen. Alle Regeln mit (*) gelten auch für $f \in L_1(\mathbb{R}^n)$ (siehe [16] §12).

Die diskrete Fouriertransformation (DFT) kann als Approximation der Fouriertransformation (FT) angesehen werden, sie steht in Zusammenhang mit dem Abtasttheorem. Dazu ein wichtiger Begriff:

Definition 1.3 *Eine Funktion $f \in \mathcal{S}(\mathbb{R}^n)$ heißt durch $b > 0$ bandbeschränkt, oder einfach $b-$bandbeschränkt, falls für alle $\xi \in \mathbb{R}^n$ mit $\|\xi\| > b$ gilt : $\hat{f}(\xi) = 0$.*

Das Abtasttheorem [1, 15] sagt aus, daß eine Funktion aus $\mathcal{S}(\mathbb{R}^n)$, die durch *b bandbeschränkt* ist, durch ihre Werte über einem kartesischen Gitter mit maximaler Kantenlänge π/b exakt rekonstruierbar und eindeutig bestimmt ist. Genauer: durch alle Abtastwerte über dem Gitter $h\,\mathbb{Z}^n$, $h \leq \frac{\pi}{b}$. Eine mathematisch konkrete Form des Abtasttheorems folgt in Kapitel 4 .
Verschwindet nun $f \in C_c^\infty(\mathbb{R}^n)$ außerhalb eines Kreises mit Radius 1, so ist die Fouriertransformierte $\hat{f}$ durch 1 *bandbeschränkt*. Die Fourierentwicklung von f ergibt eine in allen Argumenten 2-periodische Funktion $f_p \in C^\infty(\mathbb{R}^n)$. f_p ist auf $[-1, 1]^n$ identisch mit f, und es gilt:

$$f_p(x) = \left(\frac{\pi}{2}\right)^{n/2} \sum_{k \in Z^n} \hat{f}(\pi k)\, e^{i\pi <x,k>}$$

Beschränkt man sich nun im *Fourierraum* auf $(2q)^n$ Abtastwerte über dem Gitter $\pi\mathbb{Z}^n$, so ist im Fall $n = 1$ die 1-dimensionale DFT beschrieben durch:

$$\hat{f}_k = (2\pi)^{-\frac{1}{2}} \quad \frac{1}{q} \sum_{l=-q}^{q-1} f_l\, e^{-i\pi lk/q} \tag{1.1}$$

$-q \leq k < q$, $f_l := f(l/q)$, $\hat{f}_k$ ist Approximation von $\hat{f}(k\pi)$.

Indem man sich auf $2q$ Abtastwerte im Fourierraum beschränkt, begrenzt man die Auflösung im Ortsraum. Insgesamt führt die Beschränkung auf eine endliche Anzahl äquidistanter Abtastwerte im *Ortsraum* unter Benutzung des Abtasttheorems zu einer Beschränkung auf die gleiche Anzahl äquidistanter Abtastwerte im *Fourierraum* und umgekehrt.
Die Abtastwerte im Ortsraum werden durch die inverse DFT zurückgewonnen :

$$f_l = \left(\frac{\pi}{2}\right)^{1/2} \sum_{k=-q}^{q-1} \hat{f}_k \, e^{i\pi kl/q}$$

$-q \leq l < q$.

Analog zu (1.1) wird im 2-dimensionalen Fall über die kartesischen Gitterpunkte summiert:

$$\hat{f}_{k_1 k_2} = (2\pi)^{-1} \quad \frac{1}{q^2} \sum_{l_1=-q}^{q-1} \sum_{l_2=-q}^{q-1} f_{l_1 l_2} \, e^{-i\pi<k,l>/q} \tag{1.2}$$

mit $k = (k_1, k_2)$, $l = (l_1, l_2)$, $-q \leq k_1, k_2 < q-1$, $f_{l_1 l_2} := f(l_1/q, l_2/q)$, $\hat{f}_{k_1 k_2}$ ist Approximation von $\hat{f}(k_1\pi, k_2\pi)$.
Gleichung (1.2) läßt sich umschreiben (Separabilität des Kernes der FT) zu:

$$\hat{f}_{k_1 k_2} = (2\pi)^{-1} \quad \frac{1}{q^2} \sum_{l_1=-q}^{q-1} e^{-i\pi k_1 l_1/q} \sum_{l_2=-q}^{q-1} e^{-i\pi k_2 l_2/q} f_{l_1 l_2} \tag{1.3}$$

Die 2-dimensionale DFT ist durch diese Umformung auf 4q eindimensionale DFT zurückführbar.

Der FFT - Algorithmus

Zur effektiven Berechnung von (1.1) betrachte

$$A_k = \sum_{l=0}^{N-1} a_l \, e^{i2\pi lk/N} \tag{1.4}$$

$0 \leq k < N$, mit gegebenen Werten a_l.
Die Berechnung der A_k kann als Auswertung des komplexen Polynoms

$$A(x) = a_{N-1}x^{N-1} + a_{N-2}x^{N-2} + \cdots + a_1 x + a_0$$

an den Stellen ω^0, $\omega^1 = e^{2\pi i/N}$, $\omega^2, \ldots, \omega^{N-1}$ aufgefaßt werden.
Sei $N = 2^m = 2n$, $n, m \in \mathbb{N}$, also N gerade.
ω ist primitive $N-$te Einheitswurzel und es gilt: $\omega^{k+n} = -\omega^k \quad \forall k \in \mathbb{N}$.
ω^2 ist primitive $n-$te Einheitswurzel. Man sortiert nun die Koeffizienten des komplexen Polynoms in solche mit geraden und in solche mit ungeraden Indizies:

$$\begin{aligned} A(x) &= a_{N-1}x^{N-1} + a_{N-3}x^{N-3} + \cdots + a_1 x \qquad (ungerade) \\ &+ a_{N-2}x^{N-2} + a_{N-4}x^{N-4} + \cdots + a_2 x^2 + a_0 \qquad (gerade) \end{aligned}$$

Substituiert man nun $y = x^2$, so erhält man:

$$\begin{aligned} A(x) &= (a_{N-1}y^{n-1} + a_{N-3}y^{n-3} + \cdots + a_1)x \\ &+ a_{N-2}y^{n-2} + a_{N-4}y^{n-4} + \cdots + a_2 y + a_0 \end{aligned}$$

Definiert man nun zwei neue komplexe Polynome vom Grad n durch:

$$C(x) := \sum_{i=0}^{n-1} c_i\, x^i \qquad c_i = a_{2i+1}$$

$$B(x) := \sum_{i=0}^{n-1} b_i\, x^i \qquad b_i = a_{2i}$$

läßt sich $A(x)$ ausdrücken durch $A(x) = C(y)x + B(y)$.
Insgesamt erhält man :

$$\begin{aligned} A_k &= A(\omega^k) = C(\omega^{2k})\omega^k + B(\omega^{2k}) \\ A_{k+n} &= A(\omega^{k+n}) = -C(\omega^{2k})\omega^k + B(\omega^{2k}) \end{aligned}$$

$0 \leq k < n$. Somit läßt sich die Berechnung der A_k in (1.4) auf die Berechnung von

$$C_k := C(\omega^{2k}) = \sum_{l=0}^{n-1} c_l\, e^{i2\pi lk/n}$$

und

$$B_k := B(\omega^{2k}) = \sum_{l=0}^{n-1} b_l\, e^{i2\pi lk/n}$$

zurückführen, $k = 0, \ldots, n-1$.
Ist nun $N = 2^m$, so können auch diese Polynome wieder in oben beschriebener Weise geteilt werden und so fort. Durch rekursive Implementation kann man so den Rechenaufwand $O(N^2)$ für die Auswertung der A_k in (1.4) auf $O(N \log_2 N)$ reduzieren [17].

Möchte man statt der A_k wie in (1.4) α_k berechnen, für die gilt:

$$\alpha_k = \sum_{l=-q}^{q-1} a_{l+q}\, e^{i\pi lk/q} \qquad k = -q, \ldots, q-1; \quad q = N/2;$$

so setzt man unter Beachtung von $A_k = A_{k+N}\ \forall k \in \mathbb{Z}$ nach Berechnung der A_k mit obigem Algorithmus einfach $\alpha_k = (-1)^k A_k,\ k = -q, \ldots, q-1$.

Eine ausführliche Darstellung solcher FFT - Algorithmen findet sich in [10]. Dort sind auch Spezialalgorithmen für die Berechnung von DFT kleiner Längen zu finden. Diese können in einer rekursiven Implementation bei Abbruch der Rekursion verwendet werden. Zur Implementation einer FFT siehe auch [15] und [17].
Ein FFT - Algorithmus für die 2-dim. DFT kann z. B. erreicht werden, indem zuerst alle Zeilen der Eingabematrix einer 1-dim. FFT unterzogen werden, danach werden alle Spalten dieses Ergebnisses transformiert. Dieses Vorgehen nutzt die Separabilität des Kernes der DFT (1.2-1.3). Echte multidimensionale FFT - Algorithmen sind sehr kompliziert und wurden hier nicht benutzt, sie werden in [14] kurz angeschnitten.

1.2 Radontransformation und Projektionssatz

Die *Radon-Transformation* [11] ordnet einer Funktion $f \in \mathcal{S}(\mathbb{R}^n)$ die Menge ihrer Integrale über Hyperebenen des $\mathbb{R}^n$ zu.

Definition 1.4 *Für $f \in \mathcal{S}(\mathbb{R}^n)$ ist die n-dimension. Radon-Transformation definiert durch :*

$$(Rf)(\theta, s) := \int_{\theta^\perp} f(s\theta + y)\, dy$$

wobei $\theta \in S^{n-1}, s \in \mathbb{R}$. $\theta^\perp$ ist das orthogonale Komplement zu θ und damit (n-1) - dimensionaler Untervektorraum des $\mathbb{R}^n$.

Für $f \in \mathcal{S}(\mathbb{R}^n)$ gilt $(Rf) \in \mathcal{S}(Z)$ auf dem Zylinder $Z = S^{n-1} \times \mathbb{R}$, insbesondere ist für jedes feste θ die Funktion $(Rf)_\theta \in \mathcal{S}(\mathbb{R})$ [1]. Es ist leicht einzusehen, daß (Rf) auf dem Zylinder Z gerade ist, also $(Rf)(\theta, s) = (Rf)(-\theta, -s)$ gilt.
Für den weiteren Verlauf ist es wichtig, sich den praktisch wesentlichen 2-dimensionalen Fall vor Augen zu führen. Die *Radon-Transformation* beschreibt dann für festes θ und s das Linienintegral von f entlang der Geraden parallel zu $\theta^\perp$ mit – durch Vorzeichen orientiertem – Abstand s zum

Ursprung.
Da $f \in \mathcal{S}(\mathbb{R}^n) \subset L_1(\mathbb{R}^n)$ existiert die Fouriertransformation, und zwischen ihr und der Radon-Transformation von f besteht der folgende als Projektionssatz oder Fourierscheibentheorem bekannte Zusammenhang:

Satz 1.1 *Sei $f \in \mathcal{S}(\mathbb{R}^n)$, $\theta \in S^{n-1}$ fest, dann gilt:*

$$(Rf)\hat{}(\theta, \sigma) = (2\pi)^{\frac{(n-1)}{2}} \hat{f}(\sigma\theta)$$

$\sigma \in \mathbb{R}$.
Dabei handelt es sich bei $(Rf)\hat{}$ um die eindimensionale Fouriertransformation bzgl. des zweiten Arguments, bei $\hat{f}$ um die n-dim. FT.

<u>Beweis</u>: Sei $\theta_2, \ldots, \theta_n$ eine orthonormale Basis des Untervektorraumes $\theta^\perp$. Betrachte die aus den Spaltenvektoren $\theta, \theta_2, \ldots, \theta_n$ gebildete orthogonale Matrix A.
Nach der Transformationsformel für Lebesgue - integrierbare Funktionen [16] gilt nun ($|\det A| = 1$):

$$\begin{aligned}
(2\pi)^{\frac{(n-1)}{2}} \hat{f}(\sigma\theta) &= (2\pi)^{-1/2} \int\limits_{\mathbb{R}^n} f(x)\, e^{-i\,<\sigma\theta, x>}\, dx \\
&= (2\pi)^{-1/2} \int\limits_{\mathbb{R}^n} f(Ax)\, e^{-i\,<\sigma\theta, Ax>}\, dx \\
&= (2\pi)^{-1/2} \int\limits_{\mathbb{R}^n} f(Ax)\, e^{-i\,<\sigma A^\perp \theta, x>}\, dx
\end{aligned}$$

Unter Verwendung des Satzes von Fubini folgt weiter :

$$(2\pi)^{\frac{(n-1)}{2}} \hat{f}(\sigma\theta) = (2\pi)^{-1/2} \int\limits_{\mathbb{R}} e^{-i\,\sigma x_1} \int\limits_{\mathbb{R}^{n-1}} f(x_1\theta + \sum_{i=2}^{n} x_i\theta_i)\, dx_2 \ldots dx_n dx_1$$

Setze nun $s = x_1$. Das innere Integral ist die Integration über $\theta^\perp$, Maßtensor ist die Einheitsmatrix.

$$\begin{aligned}
(2\pi)^{\frac{(n-1)}{2}} \hat{f}(\sigma\theta) &= (2\pi)^{-1/2} \int\limits_{\mathbb{R}} e^{-i\,\sigma s} \int\limits_{\theta^\perp} f(s\theta + y)\, dy ds \\
&= (Rf)\hat{}(\theta, \sigma)
\end{aligned}$$

□

Die Diskretisierung und Implementation dieser Formel wird als Fourierrekonstruktion bezeichnet.

1.3 Skizze einer Fourierrekonstruktion

Es sollen an dieser Stelle einige Eigenschaften der zu rekonstruierenden Funktion f festgelegt werden, die in der ganzen Arbeit ihre Gültigkeit behalten. Das zu rekonstruierende Objekt lebe im Einheitskreis. Betrachte die reellwertige Funktion $f \in C_c^\infty(\mathbb{R}^2) \subset \mathcal{S}(\mathbb{R}^2)$ mit Träger $\Omega \subseteq \{x \in \mathbb{R}^2 \mid \|x\| \leq 1\}$.

Dann ist $\hat{f} \in \mathcal{S}(\mathbb{R}^2) \subset (L_1(\mathbb{R}^2) \cap L_2(\mathbb{R}^2))$ durch 1 bandbeschränkt, da im vorliegenden Fall $\hat{f}(x){=}f(-x), x{\in}\mathbb{R}^2$ gilt. Nach dem Abtasttheorem sind somit Werte von $\hat{f}$ über dem Gitter $\pi\mathbb{Z}^2$ für eine Rekonstruktion von $\hat{f}$ ausreichend.
Man kann für die Rücktransformation von $\hat{f}$ voraussetzen, daß f im Sinne des Shannonschen Abtasttheorems bandbeschränkt ist. Dies läßt sich im diskreten Fall durch eine Filterung im *Frequenzraum* verwirklichen (siehe auch die Bemerkung zur DFT (1.1)).

Unter der Voraussetzung, daß f durch πq - bandbeschränkt ist, läßt sich die folgende Diskretisierung wählen:

$$g(\theta_j, s_l) := (Rf)(\theta_j, s_l); \quad \theta_j = (\cos\frac{j\pi}{p}, \sin\frac{j\pi}{p}); \quad s_l = \frac{l}{q};$$

$j = 0, \ldots, p-1; \quad l = -q, \ldots, q-1; \qquad p \geq \pi q,$
wobei p für die Anzahl der Abtastrichtungen, q für die Hälfte der Linienintegrale pro Richtung steht. Dies ist eine Standard-Parallel Abtastgeometrie.

Im 1. Schritt eines Rekonstruktionsalgorithmus berechnet man nun die Approximation der Fouriertransformierten von g bzgl. des zweiten Argumentes. Diese entspricht nach Satz 1.1 der FT von f auf dem jeweiligen Zentralstrahl θ_j. Insgesamt erhält man mit obiger Abtastgeomtrie $\hat{f}$ (bis auf den Faktor aus Satz 1.1) auf einem Polargitter mit radialer Schrittweite π:

$$\hat{g}_{jr} = (2\pi)^{-\frac{1}{2}} \frac{1}{q} \sum_{l=-q}^{q-1} e^{-i\pi lr/q} g(\theta_j, s_l) \tag{1.5}$$

$$\hat{g}_{jr} \approx (2\pi)^{\frac{1}{2}} \hat{f}(r\pi\theta_j) \tag{1.6}$$

$r = -q, \ldots, q-1; \quad j = 0, \ldots, p-1$. Das Verhältnis $p \geq \pi q$ stellt sicher, daß dieses Polargitter keine Zellen enthält deren Seitenlänge π überschreitet. Dies stellt eine lockere Anwendung des Abtastheorems dar.

Im 2. Schritt muß nun eine Interpolation vom Polargitter auf ein kartesisches

Gitter durchgeführt werden, um den 3. Schritt, die 2-dimensionale Fourierrücktransformation, durch den FFT -Algorithmus durchführen zu können. Der 2. Schritt ist an dieser Stelle als " Black Box " anzusehen und soll in den folgenden Kapiteln ausführlich dargestellt werden (siehe Abb. 1.1). Es sei an dieser Stelle nur bemerkt, daß es sich dabei um die FT einer reellwertigen Funktion handelt. Es ist aus Definition 1.1 ersichtlich, daß in diesem Fall $\overline{\hat{f}(\xi)} = \hat{f}(-\xi)\ \forall \xi \in \mathbb{R}^2$ gilt. Man kann sich also bei der Interpolation auf die obere Halbebene beschränken.

.

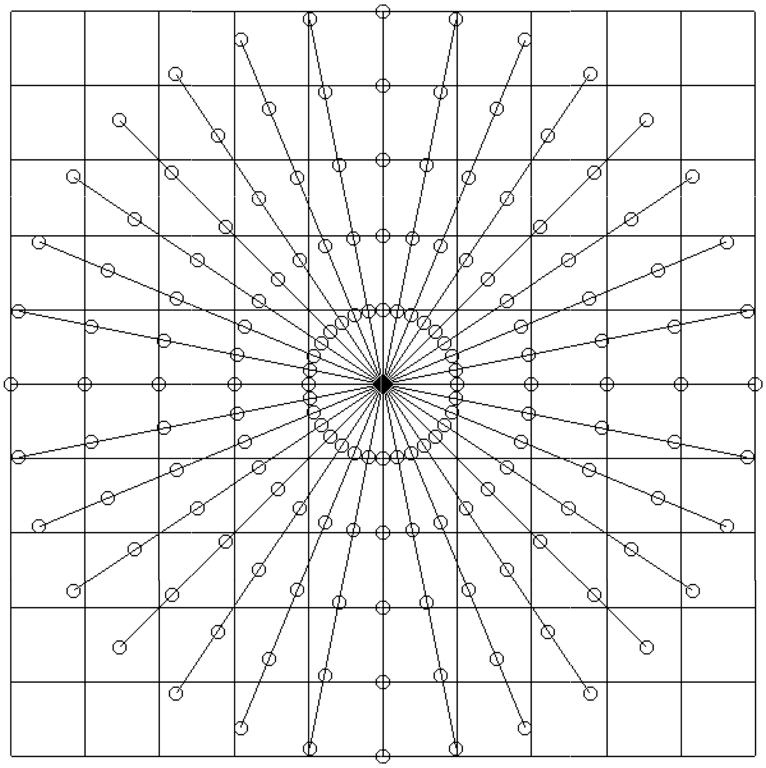


Abbildung 1.1: *Interpolation vom Polargitter auf das kartesische Gitter, p=16, q=5*

3. Schritt:
Man habe $\hat{f}_{k_1 k_2} \quad k_1, k_2 = -q, \ldots, q-1$, auf dem kartesischen Gitter $\pi\mathbb{Z}^2$, man erhält

$$f_{mn} = \frac{\pi}{2} \sum_{k_1=-q}^{q-1} e^{i\pi m k_1/q} \sum_{k_2=-q}^{q-1} e^{i\pi n k_2/q} \hat{f}_{k_1 k_2} \tag{1.7}$$

$m, n = -q, \dots, q-1$, auf dem kartesischen Gitter $\frac{1}{q}\mathbb{Z}^2$.
Man beachte an dieser Stelle, daß man sich durch die Wahl von q bei der Diskretisierung auf quadratische Details der Kantenlänge $1/q$ in der Rekonstruktion beschränkt.
Obwohl sich zeigen wird, daß der erste Schritt einem Algorithmus noch angepaßt werden muß, ist der zweite Schritt und damit die effektive Interpolation der Fourierwerte vom Polargitter auf das kartesische Gitter das eigentliche Problem.

Bemerkung : Es werden hier Funktionen aus $C_c^\infty(\mathbb{R}^2) \subset \mathcal{S}(\mathbb{R}^2)$ betrachtet, die Fouriertransformation stellt auf dem Vektorraum $\mathcal{S}(\mathbb{R}^n)$ einen Automorphismus dar. Somit existiert sie in Hin- wie auch in Rückrichtung und ist im Gegensatz zur Fouriertransformation auf $L_2(\mathbb{R}^n)$ (Satz von Plancherel [16]) in jedem Punkt eindeutig definiert. Dies ist aber keine wesentliche Einschränkung, da die Menge der C_c^∞-Funktionen im Vektorraum der Lebesgue-integrierbaren Funktionen bzgl. der L_1 -Norm dicht liegt, ebenso in L_2 bzgl. der L_2 -Norm [16].
Problematischer ist die Annahme, daß f bandbeschränkt ist. Dies trifft auf Funktionen mit kompaktem Träger nicht zu. Da $f, \hat{f} \in \mathcal{S}$, ist der Fehler, der durch diese Annahme entsteht, im Vergleich zu dem Interpolationsproblem im 2. Schritt vernachlässigbar; siehe auch in [1] den Begriff der *Quasi-Bandbeschränktheit* (essentially band-limited). Diese Ungenauigkeit läßt sich zwecks Diskretisierung auch nicht vermeiden.
Die Algorithmen werden im 6. Kapitel an Objekten getestet, die weder zu $\mathcal{S}(\mathbb{R}^2)$ gehören noch bandbeschränkt sind.

Kapitel 2

Gewöhnliche Fourier-Methoden

2.1 Die bilineare Methode

Im vorigen Kapitel wurde bereits das Problem der Interpolation vom Polargitter auf das kartesische Gitter angesprochen. Die einfachste Lösung ist sicherlich, für jeden kartesischen Punkt den jeweils nächstgelegenen polaren Punkt aufzusuchen und den entsprechenden Wert einzusetzen ("nearest-neighbour" Interpolation). In Abbildung 6.7b ist demonstriert, daß dieses Vorgehen unbrauchbare Rekonstruktionen erzeugt.
Eine etwas verfeinerte Strategie ist die sogenannte bilineare Methode [9]. Es ist schon seit langem bekannt, daß sich auch dieses Verfahren in Bezug auf die Rekonstruktionsqualität nicht mit der gefilterten Rückprojektion messen kann. Nach der Darstellung des Verfahrens soll im nächsten Abschnitt eine Erklärung für die mangelnde Leistungsfähigkeit der Methode gegeben werden.
Der erste und der dritte Schritt ist völlig analog zum Modell aus Abschnitt 1.3, man erhält also die Werte $\hat{g}_{jr}$ auf dem Polargitter

$$G_{p,q} = \{\pi r \theta_j \mid r = -q, \ldots, q-1,\ \theta_j = (\cos(\frac{j\pi}{p}), \sin(\frac{j\pi}{p})),\ j = 0, \ldots, p-1\}.$$

1. Schritt:

$$\hat{g}_{jr} = (2\pi)^{-\frac{1}{2}} \quad \frac{1}{q} \sum_{l=-q}^{q-1} e^{-i\pi lr/q} g(\theta_j, s_l)$$

$r = -q, \ldots, q-1; \quad j = 0, \ldots, p-1$.
Ziel ist nun, die Fourierwerte auf dem Gitter

$$K_q = \{\pi k \mid k \in \mathbb{Z}^2,\ \|k\| < q\}$$

durch Interpolation zu bestimmen (siehe Abb 1.1).

2. Schritt: Sei $p \geq \pi q$. Um $\hat{f}_{k_1 k_2}$ am Punkt πk, $k = (k_1, k_2)$ mit $k_2 > 0$ zu approximieren, suche die beiden πk benachbarten Zentralstrahlen θ_{j_1}, θ_{j_2}. Auf beiden Strahlen suche je 2 nächstgelegene polare Nachbarn des kartesischen Punktes. Die 4 Punkte aus Abb. 2.1 erhält man durch :

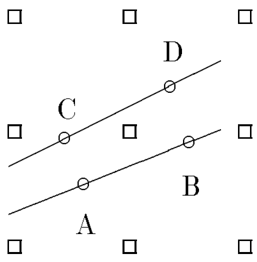


Abbildung 2.1: *Kartesischer Punkt und seine 4 polaren Nachbarn auf den beiden nächstgelegenen Zentralstrahlen*

$A = r_1 \pi \theta_{j_1}, \quad B = r_2 \pi \theta_{j_1}, \quad C = r_1 \pi \theta_{j_2}, \quad D = r_2 \pi \theta_{j_2}$

$$j_1 = \left\lfloor \frac{p}{\pi} * \arccos(\frac{k_1}{\|k\|}) \right\rfloor \qquad j_2 = j_1 + 1$$

$$r_1 = \lfloor \|k\| \rfloor \qquad r_2 = r_1 + 1.$$

Für die Abstandsverhältnisse benötigt man noch:

$$x_1 = \|k\| - r_1 \qquad x_2 = r_2 - \|k\|$$

$$x_3 = |\, j_1 \pi / p - \arccos(\frac{k_1}{\|k\|}) \,| \qquad x_4 = |\, j_2 \pi / p - \arccos(\frac{k_1}{\|k\|}) \,|$$

Die bilineare Interpolationsformel lautet :

$$\hat{f}_{k_1 k_2} = (2\pi)^{-1/2} \frac{x_4 \left(\frac{x_2 \, \hat{g}_{j_1 r_1} + x_1 \, \hat{g}_{j_1 r_2}}{x_1 + x_2} \right) + x_3 \left(\frac{x_2 \, \hat{g}_{j_2 r_1} + x_1 \, \hat{g}_{j_2 r_2}}{x_1 + x_2} \right)}{x_3 + x_4}$$

Im Falle $k_2 = 0$ erhält man die Werte natürlich ohne weitere Interpolation direkt mit Satz 1.1. Wenn p gerade ist, ebenso auf dem gesamten Achsenkreuz. Da von einem reellwertigen f ausgegangen wird, erhält man alle Werte der unteren Halbebene aus der Beziehung:

$$\hat{f}_{-k_1 - k_2} = \overline{\hat{f}_{k_1 k_2}} \qquad -q < k_1, k_2 < q$$

3. Schritt : Um die Fouriertransformierte nicht zu " hart " am Kreisrand mit Radius πq abzuschneiden, kann z. B. der cos^2-Filter verwendet werden. Hochfrequente Störungen werden so wirksam unterdrückt.

$$\hat{f}'_{k_1 k_2} = \begin{cases} \cos^2(\frac{\pi\|k\|}{2q})\hat{f}_{k_1 k_2} & : \quad \|k\| < q \\ 0 & : \quad \|k\| \geq q \end{cases}$$

$k = (k_1, k_2); \quad k_1, k_2 = -q, \ldots, q-1.$

Dieses Datenfeld wird nun der inversen 2-dim. DFT unterzogen , und man erhält die Rekonstruktion auf dem Raster $\frac{1}{q}\mathbb{Z}$:

$$f_{mn} = \frac{\pi}{2} \sum_{k_1=-q}^{q-1} e^{i\pi m k_1/q} \sum_{k_2=-q}^{q-1} e^{i\pi n k_2/q} \hat{f}'_{k_1 k_2}$$

$m, n = -q, \ldots, q-1.$

Der Rechenaufwand der bilinearen Methode ist von der Ordnung $O(q^2 \log q)$. Wie schon häufig in der Literatur beschrieben wurde, sind die Rekonstruktionsergebnisse mit dieser Methode unbefriedigend [1, 3, 4, 9]. In Abbildung 6.7c ist ein typisches Ergebnis zu sehen. Obwohl es sich um ein relativ einfaches Objekt handelt, werden nicht einmal die Funktionswerte großer Flächen zuverlässig rekonstruiert.

2.2 Bedingungen für eine Verbesserung

Eine einfache Idee zur Verbesserung der bilinearen Methode besteht darin, die Abstände der im 1. Schritt approximierten Fourierwerte zu den Punkten des kartesischen Gitters zu verringern. In den nächsten Kapiteln werden Techniken erklärt, die genau dies bewerkstelligen. Nun soll jedoch erläutert werden, daß diese Strategie auch in der Theorie erfolgversprechend ist.

In [5] wurde bereits experimentell festgestellt, daß im *Fourierraum* Interpolation in radialer Richtung kritischer ist als Interpolation in azimutaler Richtung. In [1]V.2 wird eine Bedingung (2.2) für "nearest-neighbour" Interpolation im *Fourierraum* angegeben, die in Zusammenhang mit der zu erwartenden Rekonstruktionsqualität steht (L_2-Fehler). Ferner schränkt diese Bedingung die Interpolation in radialer Richtung ein.

Da hier die Algorithmen im Vordergrund stehen, wird der Beweis der Fehlerabschätzung (Satz 2.1) nicht durchgeführt. Auch die Theorie der Sobolevräume – Unterräume des Dualraums $\mathcal{S}'(\mathbb{R}^n)$ zu $\mathcal{S}(\mathbb{R}^n)$ – soll hier nicht behandelt werden (siehe [1]). Statt dessen folgt nur die Definition der Norm des Sobolevraumes $H_0^\alpha(\Omega^n)$, wobei $\Omega^n \subset \mathbb{R}^n$ die Einheitskugel mit Mittelpunkt im Ursprung ist:

$$\|f\|_{H_0^\alpha(\Omega^n)} := \left(\int\limits_{\mathbb{R}^n} (1 + \|\xi\|^2)^\alpha \, \|\hat{f}(\xi)\|^2 \, d\xi \right)^{1/2} \qquad \alpha \in \mathbb{R} \qquad (2.1)$$

Zu jedem kartesischen Punkt πk des Gitters $\pi\mathbb{Z}^n$ sei nun ein Punkt ξ_k gegeben, sodaß gilt:

$$\| \, \xi_k - \pi k \, \| \leq h \, \|k\| \qquad (2.2)$$

mit einer von k unabhängigen Konstanten h.
Mit diesen Punkten ξ_k definiert man nun die Rekonstruktion

$$f^*(x) := \left(\frac{\pi}{2}\right)^{n/2} \sum_{\|k\| \leq a/h} e^{i\pi <x,k>} \hat{f}(\xi_k) \qquad a > 0 \qquad (2.3)$$

Der folgende Satz gibt eine Fehlerabschätzung bezüglich dieser "nearest-neighbourhood" Rekonstruktion :

Satz 2.1 *Sei $f \in C_c^\infty(\mathbb{R}^n)$ mit kompaktem Träger $\Omega \subset \Omega^n$. Die Punkte ξ_k seien im $\mathbb{R}^n$ so verteilt, daß Beziehung (2.2) erfüllt ist. Dann gibt es eine von f und h unabhängige Konstante $c(\alpha, a, n)$, sodaß für $0 \leq \alpha \leq 1$ gilt:*

$$\| \, f - f^* \, \|_{L_2(\Omega^n)} \leq c(\alpha, a, n) \, h^\alpha \, \|f\|_{H_0^\alpha(\Omega^n)}$$

Sei nun f als zu rekonstruierende Funktion fest vorgegeben.
Für die einfache "nearest-neighbourhood" Methode aus Abschnitt 2.1 ist h aus Beziehung (2.2) verhältnismäßig groß. Dies ändert sich auch nicht, wenn die Abtastgeometrie verfeinert wird. Die radiale Schrittweite bleibt π (siehe auch Abb.1.1). Satz 2.1 gibt also keinen Grund zu der Annahme, daß bei diesem Verfahren eine Vergrößerung von p und q zur Konvergenz des L_2-Fehlers gegen Null führt. Durch numerische Experimente läßt sich eine solche Annahme widerlegen. Weder die einfache "nearest-neighbourhood" Methode noch die bilineare Methode zeigt eine wesentliche Steigerung der Rekonstruktionsqualität bei proportionaler Anhebung von p und q. Die charakteristischen Störungen, die bei diesen Methoden auftreten, bleiben immer erhalten.
Von einem guten Rekonstruktionsalgorithmus sollte man verlangen, daß die Verfeinerung der Abtastgeometrie zu in jeder Hinsicht besseren Rekonstruktionen führt. Dazu ist ein Verfahren zu entwickeln, das vor Beginn einer Interpolation im *Fourierraum* Fourierwerte an Punkten ξ_k liefert, die Beziehung (2.2) mit einer möglichst kleinen Konstanten h erfüllen. h sollte dabei

mit Verfeinerung der Abtastgeometrie (p !) gegen Null streben. Die Verfahren der Kapitel 3 und 4 sind in dieser Weise konstruiert. Die Bedeutung von Satz 2.1 läßt sich durch den Vergleich der Rekonstruktionen 6.7b, 6.8a, 6.8b veranschaulichen. Die Algorithmen zu 6.8a und 6.8b erfüllen (2.2) mit $h = \pi^2/(\sqrt{2}\,p)$ bzw. $h = \pi^2/(2\,p)$.

Es soll betont werden, daß sich Satz 2.1 gegen eine einfache Interpretation sperrt, da die Konstante h in der hypothetischen Rekonstruktion (2.3) auftaucht. Er zeigt aber, daß ein Algorithmus mit kleinerem h dann eine Verbesserung darstellt, wenn die Abtastanzahl aus (2.3) entsprechend erhöht wird. Für die Algorithmen zu 6.8a und 6.8b bedeutet dies, daß man für eine Steigerung der Rekonstruktionsqualität nicht nur die Richtungszahl p erhöhen muß. Die Abtastanzahl in radialer Richtung (q !) muß entsprechend "nachrücken". Erhöht man bei diesen Verfahren p und q proportional, so konvergiert der L_2-Fehler nach Satz 2.1 gegen Null. Es soll vorweggenommen werden, daß die Fourieralgorithmen der Kapitel 3 und 4 diese erfreuliche Eigenschaft besitzen.
Die in diesem Abschnitt angestellten Überlegungen treffen nicht auf den Algorithmus des 5. Kapitels zu. Der dort dargestellte UFR-Algorithmus verfolgt im 2. Schritt eine Strategie, die eine gesonderte Betrachtung erfordert.

Kapitel 3

Der Algorithmus von Pasciak

Die eigentliche Strategie dieses Verfahrens ist es, den ersten Schritt (1.5) so abzuwandeln, daß die zu erhaltenden Fourierwerte $\hat{g}_{jr}$ auf den Linien des kartesischen Gitters liegen.
Setzt man in Abhängigkeit vom Winkel im *Fourierraum* die Schrittweite $\pi c(j)$ fest, $c(j) = 1/max\{|\sin\varphi_j|, |\cos\varphi_j|\}, \varphi_j = j\pi/p$, so liegen für $p/4 \leq j \leq 3p/4$ die Fourierwerte auf den horizontalen, für die anderen j auf den vertikalen Linien des Gitters (Abb. 3.1: $p = 12 \quad q = 3$).

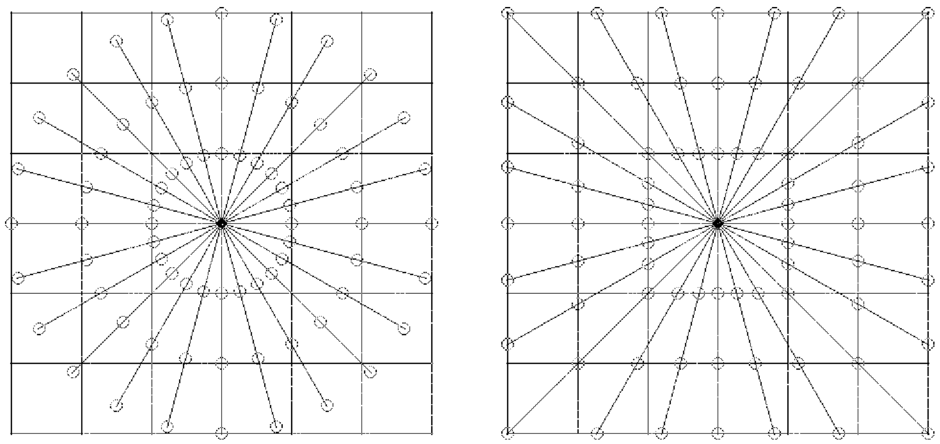


Abbildung 3.1: *Fourierwerte nach Beendigung des 1. Schrittes bei gewöhnlichen Verfahren (links), bei Pasciak (rechts)*

Benötigt wird also ein FFT -Algorithmus mit beliebiger Schrittweite im *Fre-*

quenzraum, also zur Auswertung von:

$$\hat{g}_{jr} = (2\pi)^{-\frac{1}{2}} \quad \frac{1}{q} \sum_{l=-q}^{q-1} e^{-i\pi l c(j) r/q} g(\theta_j, s_l) \tag{3.1}$$

$r = -q, \ldots, q-1; \quad j = 0, \ldots, p-1$.
Dies wird vom "Chirp-Z" Algorithmus [10] geleistet, der im folgenden beschrieben wird. Der Exponent in der Summe von (3.1) läßt sich schreiben als :

$$\frac{-i\pi l c r}{q} = \frac{i\pi c}{2q}(r-l)^2 - \frac{i\pi c}{2q}(r^2 + l^2)$$

mit $c := c(j)$. Der Richtungsindex j sei beliebig.
Betrachte (3.1) nun als

$$\hat{g}_r = e^{-i(\pi c/2q) r^2} \left((2\pi)^{-\frac{1}{2}} \frac{1}{q} \sum_{l=-q}^{q-1} e^{i(\pi c/2q)(r-l)^2} \left(e^{-i(\pi c/2q) l^2} g(\theta_j, s_l) \right) \right) \tag{3.2}$$

$r = -q, \ldots, q-1$. In dieser Form ist die Summe für festes j als diskrete Faltung erkennbar.
Setze $\hat{h}_r := \hat{g}_{jr}, \quad h_l := g(\theta_j, s_l)$, betrachte das Formeltripel

$$\begin{aligned} h'_l &= e^{-i(\pi c/2q) l^2} h_l \qquad l = -q, \ldots, q-1 \\ \hat{h}'_r &= (2\pi)^{-\frac{1}{2}} \quad \frac{1}{q} \sum_{l=-q}^{q-1} e^{i(\pi c/2q)(r-l)^2} h'_l \\ \hat{h}_r &= e^{-i(\pi c/2q) r^2} \hat{h}'_r \qquad r = -q, \ldots, q-1 \end{aligned}$$

Die zweite Formel stellt eine diskrete Faltung der Länge $2q$ dar. Um den Faltungssatz [1, 14] anwenden zu können, werden die zu faltenden Vektoren auf die Länge $4q$ gebracht, bzw. $2q$-periodisch ergänzt. Dadurch wird die Faltung zyklisch und kann als Multiplikation im *Frequenzraum* durchgeführt werden.

1. Vektor : $\underline{h'} := (h'_{-q}, h'_{-q+1}, \ldots, h'_{q-1}, 0, \ldots, 0)$

$$h'_l = 0 \text{ für } l \geq q; \quad l = -q, \ldots, 3q-1.$$

2. Vektor: $e' := (e'_{-2q}, e'_{-2q+1}, \ldots, e'_0, e'_1, \ldots, e'_{2q-1})$

$$e'_l = e^{i(\pi c/2q) l^2}; \quad l = -2q, \ldots, 2q-1.$$

$\underline{h'}$ wird mit Nullen aufgefüllt, die erste Komponente von e' : (e'_{-2q}) hat auf die weitere Rechnung keinen Einfluß und ist beliebig.
$\underline{h'}$ und e' seien nun von $-2q$ bis $2q$ indiziert, man beachte $\underline{h'}_{-2q} = h'_{-q}$ usw., sonst Komponenten wie oben dargestellt :

$$\begin{aligned} H'_m &:= \sum_{n=-2q}^{2q-1} e^{-i\pi mn/(2q)} \underline{h'}_n \\ E'_m &:= \sum_{n=-2q}^{2q-1} e^{-i\pi mn/(2q)} e'_n \\ H_m &:= H'_m E'_m \qquad m = -2q, \ldots, 2q-1 \end{aligned}$$

$$\hat{h}'_r = (2\pi)^{-\frac{1}{2}} \quad \frac{1}{4q^2} \sum_{m=-2q}^{2q-1} e^{i\pi m(r-q)/(2q)} H_m \qquad r = -q, \ldots, q-1$$

Damit ist die Faltung durch drei FFT der Länge $4q$ ersetzbar. Nach dem letzten Schritt des Formeltripels erhält man die $\hat{h}_r$, $r = -q, \ldots, q-1$.
Zu bemerken ist noch, daß es sich bei (3.1) um die FT einer reellen Funktion handelt. Demnach gilt [14]:

$$\hat{h}_{-q+r} = \overline{\hat{h}}_{q-r} \qquad r = 1, \ldots, q; \qquad \hat{h}_0 \in \mathbb{R}.$$

Diese Rechnung wird für alle Richtungen j durchgeführt und man erhält die Fourierwerte $\hat{g}_{jr}$ wie in Abb. 3.1 angedeutet.
Dieser erste Schritt des Verfahrens ist zugleich der aufwendigste. Man beachte aber, daß der 2. Vektor und damit eine Fouriertransformation objektunabhängig ist und nur von der Abtastgeometrie bestimmt wird.

Im 2. Schritt interpoliert man nun die Werte von $\hat{f}$ auf dem kartesischen Gitter $\pi\mathbb{Z}^2$, wozu man natürlich eine einfache "nearest-neighbourhood" Methode bzgl. der $\hat{g}_{jr}$ verwenden könnte, aber es stellte sich heraus, daß eine lineare Interpolation zwischen den beiden $\hat{f}_{k_1k_2}$ nächstgelegenen Punkten $\hat{g}_{j_1r}$ und $\hat{g}_{j_2r}$ erheblich bessere Rekonstruktionsergebnisse liefert. Diese Methode soll nun implementationstauglich erläutert werden:
Sei nun p mit $p \geq \pi q$ durch 4 teilbar, dies erwies sich sowohl für die Rekonstruktiongüte als auch für die Implementation als vorteilhaft, da Symmetrien ausgenutzt werden können (Beachte weiterhin einleitende *rechte* Abb. 3.1, insbesondere die Übereinstimmung auf den Diagonalen). Man will nun $\hat{f}_{k_1k_2}$ am Punkt $\pi k, k = (k_1, k_2)$ mit $\|k\| < q$ approximieren.

1.Fall: $k_1 k_2 = 0$ $\qquad \overline{k} = k_1 + k_2$

$$j_1 = \begin{cases} 0 & : \quad k_2 = 0 \\ p/2 & : \quad sonst \end{cases}$$

Setze einfach

$$\hat{f}_{k_1 k_2} = (2\pi)^{-1/2} \quad \hat{g}_{j_1 \overline{k}}$$

2.Fall: $|k_1| = |k_2| > 0$

$$j_1 = \begin{cases} p/4 & : \quad k_1 k_2 > 0 \\ 3p/4 & : \quad k_1 k_2 < 0 \end{cases}$$

Setze einfach

$$\hat{f}_{k_1 k_2} = (2\pi)^{-1/2} \quad \hat{g}_{j_1 k_2}$$

3.Fall: $|k_1| > |k_2| > 0$

$$\overline{k_1} = \begin{cases} k_1 & : \quad k_1 k_2 > 0 \\ -k_1 & : \quad k_1 k_2 < 0 \end{cases}$$

Wähle $\jmath_1, \jmath_2 \in \{0, \ldots, p/4\}$; $\quad \jmath_1 \neq \jmath_2$;
sodaß $x_i = |\,|k_2| - |k_1| c(\jmath_i) \sin(\jmath_i \pi/p)\,|$; $\quad i = 1, 2$; so klein wie möglich.

$$j_1 = \begin{cases} \jmath_1 & : \quad k_1 k_2 > 0 \\ p - \jmath_1 & : \quad k_1 k_2 < 0 \end{cases}$$

$$j_2 = \begin{cases} \jmath_2 & : \quad k_1 k_2 > 0 \\ p - \jmath_2 & : \quad k_1 k_2 < 0 \end{cases}$$

Setze nun

$$\hat{f}_{k_1 k_2} = (2\pi)^{-1/2} \quad \frac{x_2 \hat{g}_{j_1 \overline{k_1}} + x_1 \hat{g}_{j_2 \overline{k_1}}}{x_2 + x_1}$$

4.Fall: $|k_2| > |k_1| > 0$

Wähle $\jmath_1, \jmath_2 \in \{p/4, \ldots, p/2\}$; $\quad \jmath_1 \neq \jmath_2$;
sodaß $x_i = |\,|k_1| - |k_2| c(\jmath_i)| \cos(\jmath_i \pi/p)|\,|$; $\quad i = 1, 2$; so klein wie möglich.

$$j_1 = \begin{cases} \jmath_1 & : \quad k_1 k_2 > 0 \\ p - \jmath_1 & : \quad k_1 k_2 < 0 \end{cases}$$

$$j_2 = \begin{cases} \jmath_2 & : \quad k_1 k_2 > 0 \\ p - \jmath_2 & : \quad k_1 k_2 < 0 \end{cases}$$

Setze nun

$$\hat{f}_{k_1 k_2} = (2\pi)^{-1/2} \quad \frac{x_2 \hat{g}_{j_1 k_2} + x_1 \hat{g}_{j_2 k_2}}{x_2 + x_1}$$

Der 1. Fall deckt das Achsenkreuz ab, der 2. Fall die Diagonalen. In diesen beiden Fällen ist nach dem 1. Schritt des Verfahrens keine weitere Interpolation nötig.
Im 3. Fall wird eine lineare Interpolation bzgl. der 2. kartesischen Koordinate, im 4. Fall bzgl. der 1. kartesischen Koordinate durchgeführt. Die vier Fälle nutzen nicht die Reellwertigkeit der zu rekonstruierenden Funktion f. Man kann sich natürlich in diesem Fall bei der Interpolation auf die obere Halbebene beschränken (FT hermitesch, siehe Abschnitt 1.3).
Nun folgt die in Abschnitt 1.3 angesprochene Filterung. Der $\cos^2$-Filter, der im Abschnitt 2.1 benutzt wurde, erwies sich in diesem Fall als ungeeignet. Dieser Filter hat zwar sehr angenehme visuelle Eigenschaften, bei qualitativ hochwertigen Rekonstruktionsalgorithmen wird jedoch die Hochkontrastauflösung zu sehr beschnitten. Statt dessen kann der cos-Filter verwendet werden:

$$\hat{f}'_{k_1 k_2} = \begin{cases} \cos(\frac{\pi \|k\|}{2q}) \hat{f}_{k_1 k_2} & : \quad \|k\| < q \\ 0 & : \quad \|k\| \geq q \end{cases}$$

$k = (k_1, k_2); \quad k_1, k_2 = -q, \ldots, q-1.$
Dieses Datenfeld muß nur noch der Rücktransformation unterzogen werden, und man erhält die Rekonstruktion auf dem Raster $\frac{1}{q}\mathbb{Z}$:

$$f_{mn} = \frac{\pi}{2} \sum_{k_1=-q}^{q-1} e^{i\pi m k_1/q} \sum_{k_2=-q}^{q-1} e^{i\pi n k_2/q} \hat{f}'_{k_1 k_2}$$

$m, n = -q, \ldots, q-1.$
Der Rechenaufwand für die Methode von Pasciak ist von der Ordnung $O(q^2 \log q)$.

Zum Abschluß dieses Kapitels soll noch die Konstante h bestimmt werden, mit der der Algorithmus von Pasciak die Beziehung (2.2) erfüllt. Nach dem ersten Schritt sind die Fourierwerte auf folgendem Gitter verfügbar:

$$P_{p,q} := \{\pi r c(j) \theta_j \,|\, r = -q, \ldots, q-1, \theta_j = (\cos(\frac{j\pi}{p}), \sin(\frac{j\pi}{p})), j = 0, \ldots, p-1\}$$

mit $c(j) = 1/max\{|\sin(j\pi/p)|, |\cos(j\pi/p)|\}$, p sei durch 4 teilbar.

Satz 3.1 *Für jeden Punkt πk des kartesischen Gitters πZ^2 mit $\|k\| < q$ gilt:*

$$\min_{\xi \in P_{p,q}} \| \xi - \pi k \| \leq \frac{\pi^2}{\sqrt{2}\, p} \|k\|$$

<u>Beweis</u>: Sei $k = (k_1, k_2)$ mit $k_1, k_2 \geq 0$ (1. Quadrant des kart. Gitters).
1. Fall: $k_1 \geq k_2$
πk liegt auf der Senkrechten durch den Punkt $(\pi k_1, 0)$, und zwar zwischen den Zentralstrahlen mit den Indizies $j, j+1$, wobei $j \geq 0$ und $(j+1) \leq p/4$ gilt. Die Länge der von den beiden Zentralstrahlen eingeschlossenen Strecke auf dieser Senkrechten errechnet sich durch:

$$k_1\pi(\tan((j+1)\pi/p) - \tan(j\pi/p)) = \frac{k_1\pi}{\cos(j\pi/p)} \left(\frac{\sin(\pi/p)}{\cos((j+1)\pi/p)} \right)$$

mit Hilfe der Formeln für die Summe bzw. Differenz von trigonometrischen Funktionen. Aus der Einschränkung $j \geq 0$, $(j+1) \leq p/4$ folgt nun :

$$\frac{k_1\pi}{\cos(j\pi/p)} \left(\frac{\sin(\pi/p)}{\cos((j+1)\pi/p)} \right) \leq \|k\|\pi \left(\frac{\pi/p}{1/\sqrt{2}} \right) \leq \|k\|\pi \left(\frac{\sqrt{2}\,\pi}{p} \right)$$

Die Endpunkte der abgeschätzten Strecke sind Elemente von $P_{p,q}$.
2. Fall: $k_1 < k_2$
πk liegt auf der Horizontalen durch den Punkt $(0, \pi k_2)$, und zwar zwischen den Zentralstrahlen mit den Indizies $j, j+1$, wobei $j \geq p/4$ und $(j+1) \leq p/2$ gilt. Die Länge der von den beiden Zentralstrahlen eingeschlossenen Strecke auf dieser Horizontalen errechnet sich durch:

$$k_2\pi(\cot(j\pi/p) - \cot((j+1)\pi/p)) = \frac{k_2\pi}{\sin((j+1)\pi/p)} \left(\frac{\sin(\pi/p)}{\sin(j\pi/p)} \right)$$

Aus der Einschränkung $j \geq p/4$, $(j+1) \leq p/2$ folgt nun :

$$\frac{k_2\pi}{\sin((j+1)\pi/p)} \left(\frac{\sin(\pi/p)}{\sin(j\pi/p)} \right) \leq \|k\|\pi \left(\frac{\pi/p}{1/\sqrt{2}} \right) \leq \|k\|\pi \left(\frac{\sqrt{2}\,\pi}{p} \right)$$

Die Endpunkte der abgeschätzten Strecke sind Elemente von $P_{p,q}$.

In den anderen Quadranten des kartesischen Gitters folgt die Behauptung aus Symmetriegründen (p gerade).

$\square$

Der Beweis enthält abermals die trigonometrischen Zutaten des 2. Schrittes.

Kapitel 4

Der Algorithmus von Löw und Natterer

4.1 Die Methode von Löw und Natterer

Diese Strategie ist aufwendiger als das Verfahren von Pasciak, sie liefert aber bessere Rekonstruktionen. Es soll zuerst eine einfache Version erläutert werden, um dann die für den Algorithmus wesentliche *Sinc*-Interpolation durch die schnellkonvergente *Sinc*-Interpolation zu ersetzen [2].

Definition 4.1 *Für $x \in \mathbb{R}$ ist die Funktion $sinc : \mathbb{R} \to \mathbb{R}$ definiert durch:*

$$sinc\,(x) := \begin{cases} \dfrac{\sin(x)}{x} & : \quad x \neq 0 \\ 1 & : \quad x = 0 \end{cases}$$

Für $x \in \mathbb{R}^n$ ist $sinc : \mathbb{R}^n \to \mathbb{R}$ definiert durch:

$$sinc\,(x) := \prod_{i=1}^{n} sinc\,(x_i)$$

Die Bedeutung der $sinc-$Funktion liegt darin, daß sie die Fouriertransformierte der charakteristischen Funktion des Einheitsquaders im $\mathbb{R}^n$ ist. Sei also χ die charakteristische Funktion von $[-1,1]^n \subset \mathbb{R}^n$, dann gilt [1]:

$$\hat{\chi}(\xi) = (\pi/2)^{-n/2}\, sinc\,(\xi)\,, \quad \xi \in \mathbb{R}^n, \qquad \hat{\chi} = \tilde{\chi}$$

Satz 4.1 (Abtasttheorem von Shannon) *Sei $f \in L_2(\mathbb{R}^n)$ durch $b > 0$ bandbeschränkt,*
d.h. $\hat{f}(\xi) = 0$ für $\|\xi\| > b$.

Dann ist f eindeutig durch die Werte $f(hk), k \in \mathbb{Z}^n, 0 < h \leq \frac{\pi}{b}$ bestimmt, und mit Konvergenz in $L_2(\mathbb{R}^n)$ gilt :

$$f(x) = \sum_{k \in Z^n} f(hk)\, sinc\,(\frac{\pi}{h}(x - hk))$$

<u>Beweis</u>: Die Fouriertransformation ist hier als Automorphismus über dem Hilbertraum $L_2(\mathbb{R}^n)$ zu verstehen (Satz von Plancherel [16]). Aus der Bandbeschränktheit folgt $\hat{f} \in L_1(\mathbb{R}^n) \cap L_2(\mathbb{R}^n)$, somit läßt sich f nach evtl. Abänderung auf einer Nullmenge als stetig identifizieren.
Die Fourierreihe von $\hat{f}$ in $L_2([-\pi/h, \pi/h]^n)$ ist:

$$\hat{f}(\xi) = \sum_{k \in Z^n} f_k\, e^{-ih<\xi,k>}$$

$$f_k = (2\pi)^{-n}\, h^n \int\limits_{[-\frac{\pi}{h}, \frac{\pi}{h}]^n} \hat{f}(\xi)\, e^{ih<\xi,k>} d\xi$$

Da $\hat{f}$ außerhalb von $[-\pi/h, \pi/h]^n$ verschwindet, gilt für die Fourierkoeffizienten $f_k = f(hk) \cdot (2\pi)^{-n/2}\, h^n$.
Sei

$$\chi_{\frac{h}{\pi}}(\xi) := \begin{cases} 1 & : \quad \xi \in [-\pi/h, \pi/h]^n \\ 0 & : \quad sonst \end{cases}$$

Daraus folgt

$$\hat{f}(\xi) = (2\pi)^{-n/2}\, h^n \sum_{k \in Z^n} f(hk)\, \chi_{\frac{h}{\pi}}(\xi)\, e^{-ih<\xi,k>}$$

Diese Reihe konvergiert in $L_2(\mathbb{R}^n)$. Da die Fouriertransformation und ihre Inverse stetig auf $L_2(\mathbb{R}^n)$ operieren (Plancherel) gilt :

$$f(x) = (2\pi)^{-n/2}\, h^n \sum_{k \in Z^n} f(hk)\, \tilde{h}_k(x)$$

$$h_k(\xi) := \chi_{\frac{h}{\pi}}(\xi)\, e^{-ih<\xi,k>}$$

Es gilt [1]:

$$\tilde{\chi}_{\frac{h}{\pi}}(x) = (\frac{\pi}{2})^{-n/2}\, (\frac{\pi}{h})^n\, sinc\,(\frac{\pi}{h}\, x)$$

Aus der Eigenschaft der Fouriertransformation bei Translation folgt :

$$\tilde{h}_k(x) = (\frac{\pi}{2})^{-n/2}\, (\frac{\pi}{h})^n\, sinc\,(\frac{\pi}{h}(x - hk))$$

□

Der Inhalt des Satzes läßt sich für $f \in \mathcal{S}$ mittels des Faltungstheorems veranschaulichen, in diesem Fall folgt die punktweise Konvergenz der *Sinc*-Reihe gegen f [1]. Die Aussage des Satzes wird hier nur für $n = 1$ benötigt.

Nach dem ersten Schritt (1.5) hat man die $\hat{g}_{jr}$ auf einem Polargitter. Nach den Annahmen, wie in Abschnitt 1.3 beschrieben, ist f bandbeschränkt durch πq und $\hat{f}$ bandbeschränkt durch 1. Schränkt man nun $\hat{f}$ auf die jeweiligen Zentralstrahlen des Polargitters ein, ergeben sich eindimensionale 1-bandbeschränkte Funktionen (dies sieht man mit Hilfe von Satz 1.1, Träger von f ist der Einheitskreis). Die Werte dieser Funktionen sind auf den Punkten des Polargitters bekannt. Dieses Polargitter hat die radiale Schrittweite π. Mit Satz 4.1 lassen sich nun die Werte von $\hat{f}$ an jeder beliebigen Stelle eines Zentralstrahls des Polargitters berechnen. Um zum Beispiel $\hat{f}_{k_1 k_2}$, $k_2 > 0$, auf $\pi\mathbb{Z}^2$ zu approximieren, sucht man sich $j \in \{0, \ldots, p-1\}$ sodaß $\pi\|k\|\,(\cos\frac{j\pi}{p}, \sin\frac{j\pi}{p})$ möglichst nah an πk liegt und setzt für $\|k\| < q$:

$$\hat{f}_{k_1 k_2} = (2\pi)^{-1/2} \sum_{r=-q}^{q-1} \hat{g}_{jr}\ sinc\,(\pi(\|k\| - r)) \qquad k = (k_1, k_2)$$

Auf den Interpolationsschritt folgen dann Filterung und Rücktransformation. Leider ist der Aufwand für diese Methode $O(q^3)$, und damit ist gegenüber der gefilterten Rückprojektion nichts gewonnen. Es ist noch zu bemerken, daß der 2. Schritt vereinfacht dargestellt ist. Ähnlich wie beim Verfahren von Pasciak ist eine lineare Interpolation – genauer gesagt eine lineare Interpolation über dem Kreisbogen mit Radius $\|k\pi\|$ – für die Berechnung der $\hat{f}_{k_1 k_2}$ nötig. Sie wird in der verbesserten Version genau beschrieben .
Der folgende Satz ermöglicht mittels Überabtastung eine Beschleunigung der Konvergenz der Reihe aus Satz 4.1 (generalized sinc-series [2]), und damit einen weniger aufwendigen 2. Schritt im obigen Verfahren.

Satz 4.2 *Sei $f \in L_2(\mathbb{R}^n)$ durch $b > 0$ bandbeschränkt, und sei $0 < h < \frac{\pi}{b}$. Sei $\gamma \in C_c^\infty(\mathbb{R}^n)$ mit kompaktem Träger $\Omega \subseteq [-1,1]^n$, und es gelte*

$$\int_{\mathbb{R}^n} \gamma(x)dx = (2\pi)^{n/2}$$

Dann gilt mit Konvergenz in $L_2(\mathbb{R}^n)$:

$$f(x) = \sum_{k \in Z^n} f(hk)\ \tilde{\gamma}((\frac{\pi}{h} - b)(x - hk))\ sinc\,(\frac{\pi}{h}(x - hk))$$

<u>Beweis</u>: Die Fourierreihe von $\hat{f}$ in $L_2([-\pi/h, \pi/h]^n)$ ist wieder:

$$\hat{f}(\xi) = (2\pi)^{-n/2}\ h^n \sum_{k \in Z^n} f(hk)\ e^{-ih<\xi,k>} \qquad (*)$$

Es gilt $\hat{f}(\xi) = 0$ für $\|\xi\| > b$, mit $b < \frac{\pi}{h}$.
Da die rechte Seite von (*) die periodische Fortsetzung der Funktion $\hat{f} \mid [-\pi/h, \pi/h]^n$ ist, gilt für
$a = (\frac{\pi}{h} - b) + \frac{\pi}{h} = 2\frac{\pi}{h} - b > \frac{\pi}{h}$ die Gleichung (*) innerhalb von $[-a, a]^n$.

Wenn nun eine Funktion $\psi \in C_c^\infty(\mathbb{R}^n)$ mit Träger $\Gamma \subseteq [-a, a]^n$ den Wert 1 auf $[-b, b]^n$ hat, dann gilt in $L_2(\mathbb{R}^n)$:

$$\hat{f}(\xi) = (2\pi)^{-n/2} \, h^n \sum_{k \in Z^n} f(hk) \, \psi(\xi) \, e^{-ih<\xi,k>} \qquad (**)$$

Setze nun :

$$\psi(\xi) = (2\pi)^{-n/2} \int\limits_{[\frac{-\frac{\pi}{h}}{\frac{\pi}{h}-b}, \frac{\frac{\pi}{h}}{\frac{\pi}{h}-b}]^n} \gamma((\frac{\pi}{h} - b)^{-1} \, \xi - \eta) \, d\eta$$

Ist γ wie im Satz angegeben, so erfüllt ψ die obigen Bedingungen:
$\psi \in C_c^\infty(\mathbb{R}^n)$, ψ ist 1 auf $[-b, b]^n$, 0 in $\mathbb{R}^n \setminus]-a, a[^n$.

$$\begin{aligned}
\psi(\xi) &= (2\pi)^{-n/2} \int\limits_{\mathbb{R}^n} \gamma((\frac{\pi}{h} - b)^{-1} \, \xi - \eta) \, \chi((\frac{\pi}{h} - b) \, \frac{h}{\pi} \, \eta) \, d\eta \\
&= (\frac{\pi}{h} - b)^{-n} \, (2\pi)^{-n/2} \int\limits_{\mathbb{R}^n} \gamma((\frac{\pi}{h} - b)^{-1} \, (\xi - \eta)) \, \chi(\frac{h}{\pi} \, \eta) \, d\eta \\
&= (\frac{\pi}{h} - b)^{-n} \, (2\pi)^{-n/2} \, \gamma_{1/(\frac{\pi}{h}-b)} \, * \, \chi_{\frac{h}{\pi}}
\end{aligned}$$

mit $\gamma_{1/(\frac{\pi}{h}-b)} \, (\xi) := \gamma((\frac{\pi}{h} - b)^{-1} \, \xi)$ und $\chi_{\frac{h}{\pi}}$ wie in Satz 4.1. Nach dem Faltungssatz ergibt sich :

$$\tilde{\psi}(x) = \tilde{\gamma}((\frac{\pi}{h} - b) \, x) \, (\frac{\pi}{2})^{-n/2} \, (\frac{\pi}{h})^n \, sinc \, (\frac{\pi}{h} \, x)$$

Analog zum Beweis zu Satz 4.1 erhält man durch inverse Fouriertransformation von (**) unter Berücksichtigung der Translationsregel das Ergebnis.

□

Der Nutzen von Satz 4.2 liegt darin, daß $\tilde{\gamma}$ die inverse Fouriertransformierte einer C_c^∞ - Funktion ist und somit für $\|x\| \to \infty$ schneller als jede Potenz von $1/\|x\|$ gegen Null strebt [16]. Die Konvergenz der Reihe aus Satz 4.1 wird erheblich beschleunigt.
Eine geeignete Funktion γ ist zum Beispiel:

$$\gamma(\xi) = \begin{cases} c \, \cdot \, e^{1/(\|\xi\|^2 - 1)} & : \quad \|\xi\| < 1 \\ 0 & : \quad sonst \end{cases}$$

$$c = (2\pi)^{n/2} \left(\int\limits_{[-1,1]^n} e^{1/(\|\xi\|^2-1)}\, d\xi \right)^{-1}$$

Somit gilt $\tilde{\gamma}(0) = 1$. In [2] ist für den eindimensionalen Fall eine Approximation der Funktion $\tilde{\gamma} \in \mathcal{S}(\mathbb{R})$ angegeben. Für das jeweilige Teilintervall wurde folgende rationale Funktion benutzt:

$$Q(x) = \frac{a_3(x-m)^3 + a_2(x-m)^2 + a_1(x-m) + a_0}{b_3(x-m)^3 + b_2(x-m)^2 + b_1(x-m) + b_0}$$

m ist Mittelpunkt des Teilintervalles, Koeffizienten und Teilintervalle sind in der Tabelle aufgeführt.

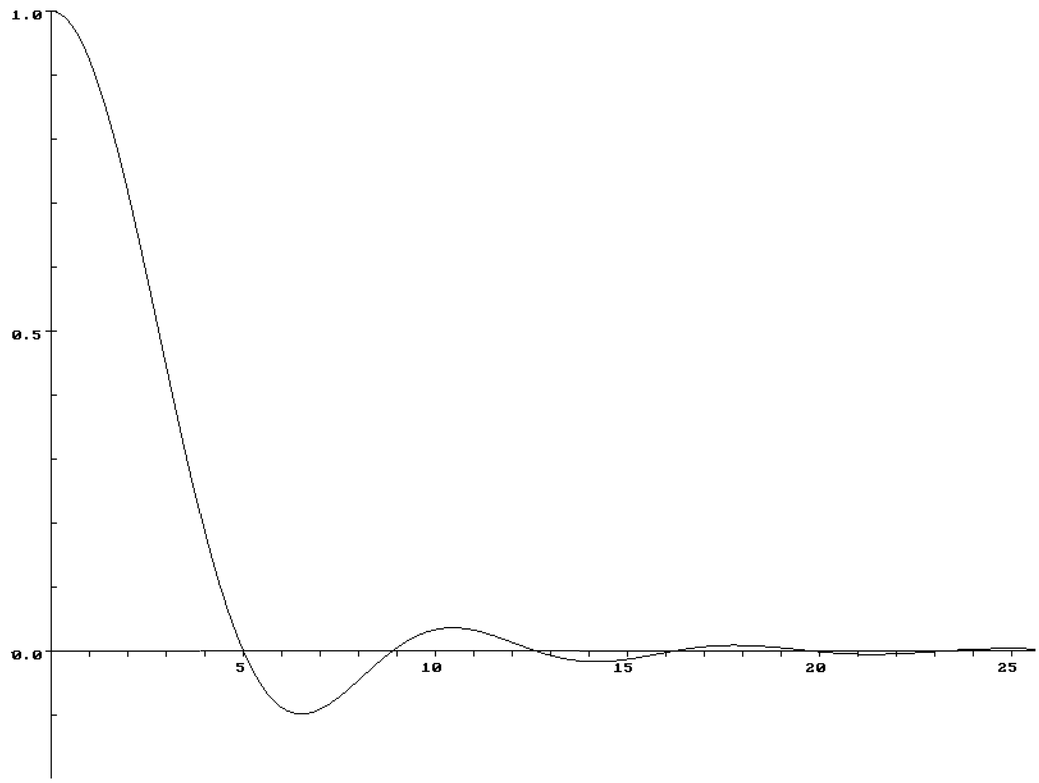
Teilintervall	a_0 b_0	a_1 b_1	a_2 b_2	a_3 b_3
[0,5]	0.584728 1.0	-0.251411 0.040010	-0.010858 0.022789	0.007110 0.000517
[5,10]	-0.071488 1.0	0.038915 0.113894	0.015432 0.034231	-0.004603 0.001948
[10,15]	0.002007 1.0	-0.020457 0.181669	-0.000241 0.057192	0.001651 0.007547
[15,20]	0.008740 1.0	0.003174 0.118630	-0.002948 0.095243	-0.000050 0.0
[20,25]	-0.002973 1.0	0.003391 0.030828	0.000693 0.048077	-0.000354 -0.002685
[25,30]	-0.001414 1.0	-0.001887 0.241197	0.000306 0.078265	0.000207 0.018285
[30,35]	0.001169 1.0	-0.000767 -0.071068	-0.000311 0.049563	0.000117 -0.012647
[35,40]	0.000257 1.0	0.000685 0.154374	-0.000060 0.066485	-0.000066 0.010572

Abbildung 4.1: *Koeffizienten der Approximation $Q(x)$ in den einzelnen Teilintervallen*

Nach [2] gilt : $|\tilde{\gamma}(x) - Q(x)| \leq 10^{-4} \quad : 0 \leq x \leq 40$

und $\quad |\tilde{\gamma}(x)| \leq 5 \cdot 10^{-4} \qquad : \; x \geq 40$

Abbildung 4.2: *Graph von $Q(x)$:*



$x \geq$	9	11	12	15	19	30
$\lvert Q(x)\rvert \leq$	0.036	0.032	0.017	0.013	0.0055	0.0015

Die Rotationssymmetrie von γ (siehe Beispiel für diese Funktion) überträgt sich auf $\tilde{\gamma}$, die Approximation Q läßt sich nur für $n = 1$ verwenden. Einen Zusammenhang zwischen rotationssymmetrischen Funktionen und ihren Fouriertransformierten liefert die Hankel-Transformation [16].

Nun zum Algorithmus : Zur Verwendung von Satz 4.2 wird im ersten Schritt des Verfahrens die Schrittweite im *Frequenzraum* halbiert (Überabtastung von $\hat{g}$ mit Faktor 2, $h = \pi/2$, $b = 1$):

$$\hat{g}_{jr} = (2\pi)^{-\frac{1}{2}} \quad \frac{1}{q} \sum_{l=-q}^{q-1} e^{-i\pi lr/(2q)} g(\theta_j, s_l)$$

$\theta_j = (\cos\frac{j\pi}{p}, \sin\frac{j\pi}{p});\ s_l = \frac{l}{q};\ r = -2q, \ldots, 2q-1;\ j = 0, \ldots, p-1$.

Dies kann folgendermaßen auf zwei FFT der Länge $2q$ reduziert werden:
Setze $\hat{h}_r := \hat{g}_{jr}, \quad h_l := g(\theta_j, s_l)$:

$$\begin{aligned}
\hat{h}_{2r} &= (2\pi)^{-\frac{1}{2}} \quad \frac{1}{q}\sum_{l=-q}^{q-1} e^{-i\pi lr/q}\, h_l \\
\hat{h}_{2r+1} &= (2\pi)^{-\frac{1}{2}} \quad \frac{1}{q}\sum_{l=-q}^{q-1} e^{-i\pi lr/q}\, e^{-i\pi l/(2q)}\, h_l
\end{aligned}$$

$r = -q, \ldots, q-1$.
Diese Rechnung wird für alle Richtungen j durchgeführt, und man erhält die Fourierwerte auf einem Polargitter mit radialer Schrittweite $\pi/2$.

2. Schritt: p mit $p \geq \pi q$ sei gerade, man möchte nun $\hat{f}_{k_1 k_2}$ am Punkt $\pi k, k = (k_1, k_2)$ mit $\|k\| < q$ approximieren.

$$\overline{k} = \begin{cases} k_1 & : \quad k_2 = 0 \\ \|k\| & : \quad k_2 > 0 \\ -\|k\| & : \quad k_2 < 0 \end{cases}$$

Wähle $\jmath_1, \jmath_2 \in \{0, \ldots, p/2\}; \quad \jmath_1 \neq \jmath_2$;
sodaß $x_i = |\ \arccos(\frac{|k_1|}{\|k\|}) - \frac{\jmath_i \pi}{p}\ |; \quad i = 1, 2$; so klein wie möglich.

$$j_1 = \begin{cases} \jmath_1 & : \quad k_1 k_2 \jmath_1 \geq 0 \\ p - \jmath_1 & : \quad k_1 k_2 \jmath_1 < 0 \end{cases}$$

$$j_2 = \begin{cases} \jmath_2 & : \quad k_1 k_2 \jmath_2 \geq 0 \\ p - \jmath_2 & : \quad k_1 k_2 \jmath_2 < 0 \end{cases}$$

Setze nun

$$\hat{f}_{k_1 k_2} = (2\pi)^{-1/2} \sum_{\pi|\overline{k} - r/2| < L} \hat{g}_r\ \tilde{\gamma}\left(\pi(\overline{k} - r/2)\right)\ sinc\left(2\pi(\overline{k} - r/2)\right)$$

mit

$$\hat{g}_r := \begin{cases} \hat{g}_{j_1 0} & : \quad r = 0 \\ \dfrac{x_2 \hat{g}_{j_1 r} + x_1 \hat{g}_{j_2 r}}{x_2 + x_1} & : \quad sonst \end{cases} \qquad r = -2q, \ldots, 2q-1.$$

Durch die lineare Interpolation zwischen den Werten $\hat{g}_{j_1 r}, \hat{g}_{j_2 r}$ auf den beiden $\pi(k_1, k_2)$ benachbarten Zentralstrahlen $t\theta_{j_1}, t\theta_{j_2}; t \in \mathbb{R}$; wird ein Zentralstrahl simuliert, der diesen kartesischen Punkt genau trifft. Auf diesen simulierten Strahl wird nun Satz 4.2 angewendet, wobei L den "vorzeitigen" Abbruch der *Sinc*-Reihe steuert. Es ist offensichtlich, daß mit L ein Kompromiß zwischen Geschwindigkeit und Genauigkeit des Verfahrens gewählt wird. L stellt hier gleichzeitig den Abbruch der schnell gegen Null strebenden Funktion $\tilde{\gamma}$ dar; vorweg soll mit $L \geq 12$ eine Orientierung gegeben werden. Dieser Wert erwies sich in den numerischen Experimenten als Minimum für die Erzeugung passabler Rekonstruktionen. Der 2. Schritt nutzt hier nicht die Reellwertigkeit von f. Die Beschränkung der Prozedur auf die obere Halbebene spart eine Menge komplexer Rechenoperationen und damit viel Zeit.

Der 3. Schritt ist mit Filterung und Rücktransformation analog zum Verfahren von Pasciak :

$$\hat{f}'_{k_1 k_2} = \begin{cases} \cos(\frac{\pi\|k\|}{2q})\hat{f}_{k_1 k_2} & : \quad \|k\| < q \\ 0 & : \quad \|k\| \geq q \end{cases}$$

$k_1, k_2 = -q, \ldots, q-1.$

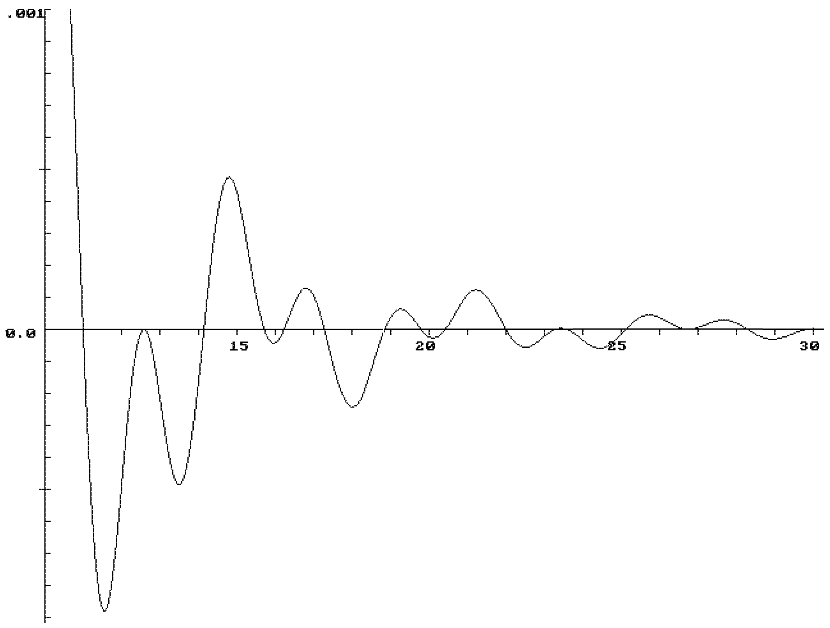
$$f_{mn} = \frac{\pi}{2} \sum_{k_1=-q}^{q-1} e^{i\pi m k_1/q} \sum_{k_2=-q}^{q-1} e^{i\pi n k_2/q} \hat{f}'_{k_1 k_2}$$

$m, n = -q, \ldots, q-1.$

Zur effektiven Implementation sei bemerkt, daß im 2. Schritt bis auf die *Sinc*-Reihe nur objektunabhängige Berechnungen durchgeführt werden. Es ist sinnvoll, Informationen über nächstgelegene Strahlen, Abstände und ein Flag, das angibt ob eine lineare Interpolation zwischen zwei Strahlen überhaupt notwendig ist ($x_1 \cdot x_2 = 0$), vor der eigentlichen Rekonstruktion abzuspeichern. Diese Informationen ändern sich erst mit der Wahl einer anderen Abtastgeometrie.
Eine Alternative zum zweiten Schritt wäre die senkrechte Projektion des kartesischen Punktes auf beide benachbarten Zentralstrahlen mittels des Skalarproduktes. Es müssen dann zwei *Sinc* -Reihen für die Fourierwerte an den Punkten $< \theta_{j_1}, k\pi > \theta_{j_1}$ und $< \theta_{j_2}, k\pi > \theta_{j_2}$ berechnet werden. Dann wird durch lineare Interpolation der Wert f_k bestimmt. Diese Variante erhöht allerdings nur den Rechenaufwand, es ist keine Verbesserung der Rekonstruktion festzustellen.
Sowohl beim Verfahren von Pasciak als auch beim obigen Verfahren würde es sich anbieten, die Filterung vor der Interpolation im zweiten Schritt durchzuführen. Die Ergebnisse sind allerdings in der Praxis nahezu identisch, Qualitätsunterschiede sind nicht festzustellen.

Die Abbruchkonstante L in der obigen Version des Verfahrens verdient eine nähere Betrachtung. Der folgende Graph (Abb. 4.3) soll eine Wahl erleichtern. Diese Funktion ist die Approximation von $\tilde{\gamma}(x) \cdot sinc(2x)$, also die hier

Abbildung 4.3: $Q(x) \cdot sinc\,(2x)$



verwendete Interpolationsfunktion aus Satz 4.2.
Für $x > 12$ ist der absolute Funktionswert kleiner als $7 \cdot 10^{-4}$, für $x > 22$ sinkt er unter 10^{-4}. In praktischen Versuchen zeigte sich, daß mit $L = 23$ eine Rekonstruktionsqualität einsetzte, die sich mit größeren L nicht mehr nennenswert steigerte.
Vorsichtshalber sollen jedoch die theoretischen Abhängigkeiten von L untersucht werden.

Korollar 4.3 *Sei $f \in L_2(\mathbb{R}^n)$ durch b bandbeschränkt, und sei $0 < h \leq \frac{\pi}{b}$. Dann gilt:*

$$\|f\|_{L_2} = \left(h^n \sum_{k \in Z^n} |f(hk)|^2 \right)^{1/2}$$

Beweis: Betrachte die Fourierreihe von $\hat{f}$ aus dem Beweis von Satz 4.1 mit Fourierkoeffizienten $f_k = f(hk) \cdot (2\pi)^{-n/2}\ h^n$.

Für $\hat{f} \in L_2([-\pi/h, \pi/h]^n)$ gilt die Parsevalsche Gleichung oder Vollständigkeitsrelation:

$$\|\hat{f}\|_{L_2}^2 = \left(\frac{2\pi}{h}\right)^n \sum_{k \in Z^n} |f_k|^2$$

Die Fouriertransformation ist Isometrie auf $L_2(\mathbb{R}^n)$. Nach dem Satz von Plancherel [16] gilt:

$$\|f\|_{L_2} = \|\hat{f}\|_{L_2}$$

□

Betrachte nun die abgebrochene *Sinc*-Reihe (Bezeichnungen wie in Satz 4.2, $n = 1$, $x \in \mathbb{R}$):

$$f_L(x) := \sum_{(\pi/h-b)|x-hk|<L} f(hk)\ \tilde{\gamma}((\frac{\pi}{h} - b)(x - hk))\ sinc\,(\frac{\pi}{h}(x - hk))$$

Satz 4.4 *Sei* $f \in L_2(\mathbb{R})$ *durch* $b > 0$ *bandbeschränkt, und sei* $0 < h < \frac{\pi}{b}$. *Sei* γ *wie in Satz 4.2 ,* $n = 1$. *Dann gilt (bis auf Nullmenge) :*

$$|f(x) - f_L(x)| \leq h^{-1/2}\ \|f\|_{L_2} \cdot \sup\{|\tilde{\gamma}(x)| : x \geq L\}$$

Bemerkung : Die Einschränkung – bis auf Nullmenge – kann für $f \in \mathcal{S}$ gestrichen werden.
Beweis: $|f(x) - f_L(x)|^2$

$$\leq \left| \sum_{(\pi/h-b)|x-hk|\geq L} f(hk)\ \tilde{\gamma}((\frac{\pi}{h} - b)(x - hk))\ sinc\,(\frac{\pi}{h}(x - hk)) \right|^2$$

$$\leq |\sup\{|\tilde{\gamma}(x)| : x \geq L\}|^2 \left| \sum_{(\pi/h-b)|x-hk|\geq L} |f(hk)|\ |sinc\,(\frac{\pi}{h}(x - hk))| \right|^2$$

Aus Korollar 4.3 ergibt sich:

$$\sum_{k \in Z} |f(hk)|^2 = h^{-1}\|f\|_{L_2}^2$$

Da *sinc* durch 1 bandbeschränkt ist, ebenso wie jede Funktion die durch Translation der *sinc* - Funktion entsteht, folgt ebenfalls mit Korollar 4.3:

$$\sum_{k \in Z} |sinc\,(\frac{\pi}{h}(x - hk))|^2 = \pi^{-1}\|sinc\|_{L_2}^2 \ = \ 1$$

unter Verwendung von

$$\int_{\mathbb{R}} \left(\frac{\sin x}{x}\right)^2 dx = \pi$$

und der Translationsinvarianz des Integrals.
Aus der Cauchy-Schwarzschen Ungleichung folgt:

$$|f(x) - f_L(x)|^2 \leq |\sup\{|\tilde{\gamma}(x)| : x \geq L\}|^2 \sum_{k \in Z} |f(hk)|^2 \sum_{k \in Z} |sinc\,(\frac{\pi}{h}(x - hk))|^2$$

□

Die Fouriertransformation einer zu rekonstruierenden Funktion $f \in C_c^\infty(\mathbb{R}^2)$ ist beschränkt durch

$$|\hat{f}(\xi)| \leq (2\pi)^{-1} \|f\|_{L_1}\,,\ \forall \xi \in \mathbb{R}^2$$

Dies läßt wiederum im *Fourierraum* eine Abschätzung der L_2-Norm der Funktion $\hat{g}$ – eingeschränkt auf einen Zentralstrahl – zu, um Satz 4.4 anzuwenden. Der maximale absolute Funktionswert von f sei M, Träger ist der Einheitskreis. Im *Fourierraum* wird nun die *Sinc*-Reihe über einen Strahl der Länge $2\pi q$ berechnet, dies zeigt eine Abhängigkeit von q und L. Es gilt:

$$\|\hat{g}_\theta\|_{L_2} \leq M\pi q^{1/2}$$

Allerdings ist diese Abschätzung sehr grob, da die Fouriertransformierte einer L_1-Funktion für $\|\xi\| \to \infty$ gegen Null strebt, für Funktionen aus C_c^∞ sogar schneller als jede Potenz von $1/\|\xi\|$.
Der Wert L=23 bezieht sich hier auf eine Abtastgeometrie mit $p = 400, q = 128$, er schien aber auch für $q = 320$ noch ausreichend zu sein. Die obige Ungleichung zeigt, daß eine Verdoppelung von q den Fehler aus Satz 4.4 höchstens um den Faktor $\sqrt{2}$ erhöht. Aufgrund des sehr schnellen Abfalls von $\tilde{\gamma}$ muß L nur etwas vergrößert werden um den Fehler auszugleichen. In der Praxis kann also L fast konstant gehalten werden, dies rechtfertigt, den Aufwand für das Verfahren von Löw-Natterer als $O(q^2 \log q)$ anzusehen.
Das Verfahren erfüllt die Bedingung (2.2) für die "nearest-neighbourhood" Rekonstruktion aus Abschnitt 2.2 mit $h = \pi^2/(2\,p)$.

4.2 Die Methode von Stark

Die Interpolation in radialer Richtung wurde im Algorithmus von Löw-Natterer mit großem Aufwand durchgeführt. Man kann sich fragen, ob eine lineare Interpolation in azimutaler Richtung angemessen ist. Diese Fragestellung führt zu einem weiteren Verfahren, der Methode von Stark [5, 6] .

Der 1. Schritt (1.5) und der 3. Schritt sind unverändert aus Abschnitt

1.3 zu entnehmen. Die Fourierwerte $\hat{g}_{jr}$ des Polargitters sollen jetzt etwas anders bezeichnet werden, definiere:

$$G_r(\varphi_j) := \begin{cases} \hat{g}_{p+j,-r} & : \quad j < 0 \\ \hat{g}_{jr} & : \quad 0 \leq j < p \\ \hat{g}_{j-p,-r} & : \quad j \geq p \end{cases}$$

$\varphi_j = j\pi/p,\ j = -p, \ldots, 2p-1,\ r = -q+1, \ldots, q-1.$
Faßt man $G_r(\varphi)$ als kontinuierliche Funktion auf, kann man diese als $2\pi-$periodisch betrachten. Sie beschreibt Kreise mit Radius $\pi|r|$ auf der Fouriertransformierten der zu rekonstruierenden Funktion f, genauer:

$$G_r(\varphi) = (2\pi)^{1/2}\, \hat{f}(r\pi(\cos\varphi, \sin\varphi))$$

In [5, 6] wird der Begriff der Winkel-Bandbeschränktheit (angularly band-limited) eingeführt. Die zu rekonstruierende Funktion f ist durch $K \in \mathbb{N}$ Winkel-bandbeschränkt, wenn für alle Radien $r\pi$ die Fourierreihe von $G_r(\varphi)$ in $[-\pi, \pi]$ durch

$$G_r(\varphi) = \sum_{k=-K}^{K} c_{rk}\, e^{ik\varphi}$$

darstellbar ist, also nur $2K+1$ Terme enthält.
Für die Fourierkoeffizienten c_{rk} gilt:

$$c_{rk} = (2\pi)^{-1} \int\limits_{[-\pi,\pi]} G_r(\varphi)\, e^{-i\varphi k} d\varphi$$

Die Abtastanzahl in azimutaler Richtung ist im Modell aus Abschnitt 1.3 $2p$. Setze nun $p = K+1$, r beliebig. Die Berechnung der c_{rk} mittels DFT ergibt:

$$c_{rk} = \frac{1}{2p} \sum_{l=-p}^{p-1} G_r(\varphi_l)\, e^{-ikl\pi/p} \qquad k = -K, \ldots, K$$

Die Grenzen der obigen Summe erklären sich folgendermaßen:
Man hat es hier mit einer 2π-periodischen Funktion zu tun, die im Intervall $[-\pi, \pi]$ mindestens $(2K+1)$ - mal abgetastet werden muß (siehe Abschnitt 1.1 und die Gesetze zur Abtastung periodischer Funktionen in [15]). Ferner ist eine gerade Anzahl von Abtastungen notwendig, damit man die Symmetrie der *Radon-Transformation* wie in der Abtastgeometrie von Abschnitt 1.3 nutzen kann. Die Messung der Linienintegrale wurde in diesem Abschnitt auf einen Winkelbereich von 0 bis π beschränkt, da g bzw. (Rf) gerade ist. In

azimutaler Richtung stehen daher für jeden Radius $2p$ Fourierwerte $G_r(\varphi_l)$ zur Verfügung.

Insgesamt ergibt sich folgende Darstellung:

$$G_r(\varphi) = \frac{1}{2p} \sum_{l=-p}^{p-1} G_r(\varphi_l) \left(\sum_{|k|<p} e^{ik(\varphi - l\pi/p)} \right)$$

Für $z \neq 0$ und $z \neq 1$ folgt aus der Summenformel für die endliche *geometrische* Reihe :

$$\sum_{|k|<p} z^k = \frac{z^p - z^{-p+1}}{z-1}$$

Setze $z = e^{i(\varphi - l\pi/p)}$ und folgere für $\varphi \in]-\pi, \pi[$ mit $\varphi \neq l\pi/p$, $\forall l \in \mathbb{Z}$:

$$G_r(\varphi) = \frac{1}{2p} \sum_{l=-p}^{p-1} G_r(\varphi_l) \frac{\sin((p-\frac{1}{2})(\varphi - l\pi/p))}{\sin(\frac{1}{2}(\varphi - l\pi/p))}$$

Mit dieser Formel läßt sich nun der 2. Schritt des Rekonstruktionsverfahrens von Stark konstruieren:
f erfülle alle Voraussetzungen aus Abschnitt 1.3 und sei durch K Winkelbandbeschränkt mit $K = p - 1$. Der 1. Schritt sei wie in (1.5) durchgeführt worden. Die Werte $G_r(\varphi_j)$ sind aus den $\hat{g}_{jr}$ bestimmt worden.
2. Schritt: Sei $p \geq \pi q$. Man möchte $\hat{f}_{k_1 k_2}$ am Punkt πk, $k = (k_1, k_2)$ mit $k_2 > 0$ approximieren.
1. Fall: Der Punkt πk liegt auf einem Zentralstrahl des Polargitters. Es existiert also $j \in \{0, \ldots, p-1\}$ mit $j\pi/p = \arccos(\frac{k_1}{\|k\|})$. Durch die *Sinc*-Reihe über diesen Strahl läßt sich der gesuchte Wert berechnen:

$$\hat{f}_{k_1 k_2} = (2\pi)^{-1/2} \sum_{r=-q}^{q-1} G_r(\varphi_j)\, sinc\,(\pi(\|k\| - r))$$

mit $\varphi_j = j\pi/p$.
2. Fall: Der Punkt πk liegt nicht auf einem Zentralstrahl des Polargitters. Setze $\varphi = \arccos(\frac{k_1}{\|k\|})$:

$$\hat{f}_{k_1 k_2} = (2\pi)^{-1/2} \sum_{r=-q}^{q-1} \left(\frac{1}{2p} \sum_{l=-p}^{p-1} G_r(\varphi_l)\, \sigma(\varphi - \varphi_l) \right) sinc\,(\pi(\|k\| - r))$$

mit $\varphi_l = l\pi/p$ und

$$\sigma(x) := \frac{\sin((p-\frac{1}{2})\,x)}{\sin(\frac{1}{2}\,x)}$$

Alle Werte der unteren Halbebene ergeben sich für eine reellwertige Funktion f aus den Werten der oberen Halbebene. Die Winkel-Bandbeschränktheit wird durch Filterung der Vektoren $(\ G_r(\varphi_{-p}), \ldots, G_r(\varphi_{p-1})\ );\ r = -q+1, \ldots, q-1;$ erzwungen [5].
Dieser Algorithmus hat einen Aufwand der Ordnung $O(q^4)$. Kombiniert man das Verfahren mit der schnellkonvergenten *Sinc*-Reihe aus Abschnitt 4.1, hat man immer noch einen Aufwand der Ordnung $O(q^3)$.

Die Methode von Stark wurde in den numerischen Experimenten konsequent durchgeführt. Sie erzeugt etwas bessere Rekonstruktionen als das Verfahren von Löw -Natterer. Dies liegt aber zum großen Teil an der Filterung in azimutaler Richtung. Der Algorithmus von Löw-Natterer kann folgendermaßen zur gleichen Qualität gebracht werden :

1. Die Filterung in azimutaler Richtung wird ebenfalls durchgeführt.

2. Die lineare Interpolation in azimutaler Richtung (2. Schritt) wird durch eine quadratische ersetzt.

Mit dieser Technik konnten in jeder Hinsicht gleichwertige Rekonstruktionen erzeugt werden (vergleiche Abb. 6.12d und 6.13a), die zusätzliche Filterung steigert aber den Rechenaufwand wesentlich. Die Verfahren dieses Abschnitts strapazieren die eigentliche Attraktivität der Fourierrekonstruktion – ihre Geschwindigkeit – in bedenklicher Weise. Es soll auch nicht verschwiegen werden, daß das ohnehin schon sehr komplizierte Löw-Natterer Verfahren durch diese Modifikation zum Alptraum eines Programmierers werden kann.

Kapitel 5

Der UFR-Algorithmus

Der UFR-Algorithmus (unified Fourier reconstruction) hat mit allen bisher beschriebenen Verfahren gemeinsam, daß die Interpolation im Gegensatz zur gefilterten Rückprojektion im *Fourierraum* durchgeführt wird. Dennoch fällt dieser Algorithmus etwas aus dem Rahmen, denn die polaren Fourierwerte werden hier in einer indirekteren Weise verwendet. Es soll zuerst eine einfache Version beschrieben werden [7], um die Problematik zu verdeutlichen. Danach wird eine modifizierte Version angegeben, die zu ausgezeichneten Rekonstruktionsergebnissen führt.
Der 1. und der 3. Schritt inklusive Filterung seien unverändert aus Abschnitt 2.1 entnommen, dem 2. Schritt liegt folgende analytische Überlegung zugrunde:

Definiere für $x \in \mathbb{R}^2$

$$I(x) = \begin{cases} 1 & : \quad \|x\| \leq 1 \\ 0 & : \quad sonst \end{cases}$$

I ist die charakteristische Funktion des Einheitskreises im $\mathbb{R}^2$, ihre Rotationssymmetrie überträgt sich auf ihre Fouriertransformierte. Es gilt [16]:

$$\hat{I}(\xi) = \begin{cases} \dfrac{J_1(\|\xi\|)}{\|\xi\|} & : \quad \xi \in \mathbb{R}^2 \setminus \{0\} \\ 0.5 & : \quad \xi = 0 \end{cases}$$

J_1 ist Besselfunktion 1. Ordnung, aufgrund der Rotationssymmetrie reicht der untenstehende Graph (Abb. 5.1) zur Erklärung von $\hat{I}$.
Sei nun $f \in C_c^\infty(\mathbb{R}^2)$ mit Träger $\Omega \subseteq \{x \in \mathbb{R}^2 \mid \|x\| \leq 1\}$ die zu rekonstruierende Funktion. Somit hat man mit I, wie mit jeder anderen charakteristischen Funktion, deren kompakter Träger den Einheitskreis überdeckt, die

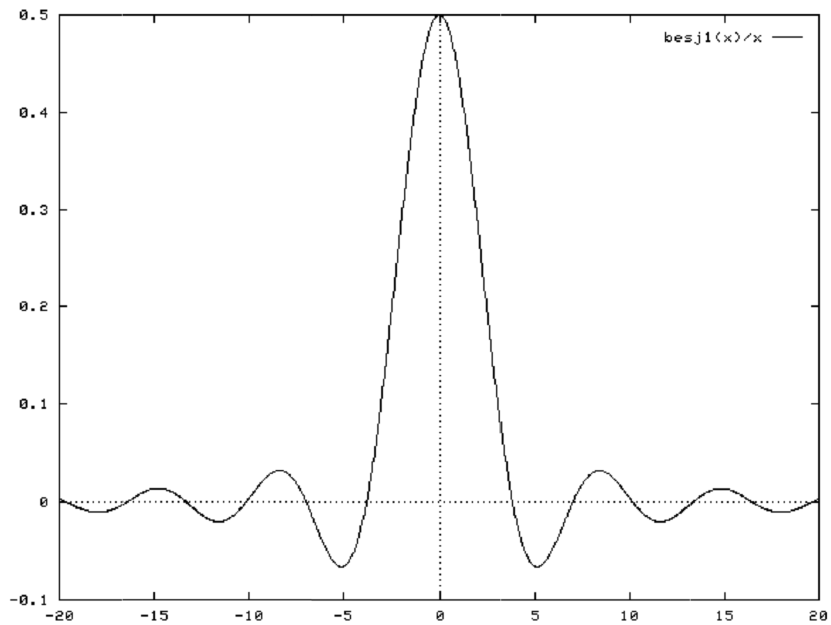


Abbildung 5.1: $J_1(x)/x \qquad x \in R$

Beziehung:

$$(f \cdot I)(x) = f(x) \qquad \forall x \in \mathbb{R}^2 \tag{5.1}$$

Es folgt nun:

$$\begin{aligned} \hat{f}(\xi) &= (2\pi)^{-1} \hat{I}(\xi) * \hat{f}(\xi) \\ &= (2\pi)^{-1} \int\limits_{\mathrm{R}^2} \hat{I}(\xi - y) \hat{f}(y) \, dy \end{aligned}$$

Damit gilt in ebenen Polarkoordinaten :

$$\begin{aligned} \hat{f}(\xi) &= (2\pi)^{-1} \int\limits_0^{2\pi} \int\limits_0^{\infty} \hat{I}(\xi - s(cos\varphi, sin\varphi)) \hat{f}(s(cos\varphi, sin\varphi)) \, s \, ds \, d\varphi \\ &= (2\pi)^{-1} \int\limits_0^{\pi} \int\limits_0^{\infty} \hat{I}(\xi - s(cos\varphi, sin\varphi)) \hat{f}(s(cos\varphi, sin\varphi)) \, s \, ds \, d\varphi \\ &+ (2\pi)^{-1} \int\limits_0^{\pi} \int\limits_0^{\infty} \hat{I}(\xi + s(cos\varphi', sin\varphi')) \hat{f}(-s(cos\varphi', sin\varphi')) \, s \, ds \, d\varphi' \\ & mit \quad \varphi' = \varphi + \pi \end{aligned}$$

$$= \quad (2\pi)^{-1} \int_0^\pi \int_{-\infty}^\infty \hat{I}(\xi - s(cos\varphi, sin\varphi))\hat{f}(s(cos\varphi, sin\varphi))\,|s|\,ds\,d\varphi$$

Setze nun entsprechend zu Abschnitt 1.3: $g(\theta, s) := (Rf)(\theta, s)$. Nach Satz 1.1 gilt:

$$\hat{f}(s\theta) = (2\pi)^{-\frac{1}{2}}\,\hat{g}(\theta, s) \qquad s \in \mathbb{R}$$

mit $\theta = (cos\varphi, sin\varphi) \in S^1$ fest.
Setzt man nun $G(\varphi, s) := \hat{g}((cos\varphi, sin\varphi), s)$; $s \in \mathbb{R}, \varphi \in [0, \pi]$; so läßt sich schreiben:

$$\hat{f}(\xi) = (2\pi)^{-\frac{3}{2}} \int_0^\pi \int_{-\infty}^\infty \hat{I}(\xi - s(cos\varphi, sin\varphi))G(\varphi, s)\,|s|\,ds\,d\varphi \tag{5.2}$$

Nach dem 1. Schritt (1.5) hat man die Werte $\hat{g}_{jr}$ auf dem Polargitter mit radialer Schrittweite π. Dann stellt sich das Problem der Diskretisierung und Implementation von (5.2). Dazu wird in [7, 8] folgende Strategie angegeben:

$$\hat{f}_{k_1k_2} = (2\pi)^{-\frac{3}{2}} \sum_{j=0}^{p-1} \sum_{r=-q}^{q-1} \hat{I}\left(\pi(k_1, k_2) - r\pi(cos\frac{j\pi}{p}, sin\frac{j\pi}{p})\right)\,\hat{g}_{jr}\,J(j, r) \tag{5.3}$$

$k_1, k_2 = -q, \ldots, q-1$.
Der Skalierungsfaktor $J(j, r)$ verteilt hierbei Gewichte über den Punkten des Polargitters :

$$J(j, r) = |\pi r|\,(\pi/p)\pi \tag{5.4}$$

Natürlich ist eine Berechnung von (5.3) zu aufwendig, denn dies würde zu einem Algorithmus mit einem Aufwand der Ordnung $O(q^4)$ führen. Die Summe muß auf eine Menge $\pi(k_1, k_2)$ benachbarter Punkte eingeschränkt werden. Dazu werden für die Approximation von $\hat{f}_{k_1k_2}$ mit (5.3) Indizies (j, r) gewählt, sodaß $\pi(k_1, k_2)$ einer der L nächsten Nachbarn von $\pi r(cos\ j\pi/p, sin\ j\pi/p)$ ist. In der Implementation durchläuft man das Polargitter, sucht jeweils die L nächsten kartesischen Nachbarn auf und addiert jeweils das Produkt aus (5.3) zu dem bisherigen Wert des jeweiligen kartesischen Punktes (siehe Abb. 5.2). Diese Strategie (data–driven) wurde in [7] vorgeschlagen und ist etwas verwirrend. Die Suche nach nächstgelegenen polaren Punkten beim Durchlaufen des kartesischen Gitters (grid-driven) wäre hingegen weniger implementationsfreundlich.
In [7] wird $L = 9$ angegeben, was einem Abschneiden der rotationssymmetrischen Interpolationsfunktion $\hat{I}$ bei ca. π entspricht. Dies erscheint aufgrund des verhältnismäßig langsamen Abfalls von $\hat{I}$ riskant. In der Tat stellen sich bei allen Rekonstruktionen mit dieser Version des Verfahrens

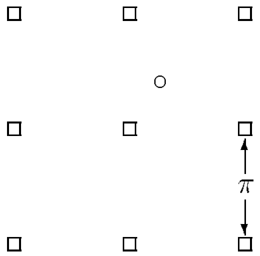


Abbildung 5.2: *Datenpunkt des Polargitters und seine 9 kartesischen Nachbarn*

großflächige Bildfehler ein, deren Intensität mit wachsendem L abnimmt. Der Rechenaufwand wird dadurch erheblich heraufgesetzt. Es ist besser, I aus (5.1) durch eine Funktion zu ersetzen, deren Fouriertransformation sehr viel schneller fällt, damit L möglichst klein gehalten werden kann.
Bei Benutzung der Integrationsformel (5.3) treten noch weitere Probleme auf. Wie schon angesprochen wurde, verteilt J in (5.3) Gewichte, abhängig vom Radius, über allen Punkten des Polargitters. Es ist wichtig zu bemerken, daß die Genauigkeit einer solchen Konstruktion mit Verfeinerung der Abtastgeometrie $(p,\, q)$ nicht wesentlich steigt. Die radiale Schrittweite ist im *Frequenzraum* konstant. Dies war wesentliches Problem der bilinearen Methode (siehe Abschnitt 2.2). Wenn man es aber mit dem Produkt bandbeschränkter Funktionen zu tun hat (siehe auch [1], Parsevalsche Beziehung), kann man hoffen, daß eine derartige Integrationsformel ab einer gewissen Feinheit des Polargitters gute Ergebnisse liefert. Die radiale Schrittweite im *Frequenzraum* kann mit einfachen Techniken, wie schon in Kapitel 4 benutzt, herabgesetzt werden. Ferner muß die Interpolationsfunktion $\hat{I}$ so ersetzt werden, daß ihr Träger durch die Stützstellen der numerischen Integration in (5.3) " gut " getroffen wird und sie einer gewissen *Quasi-Bandbeschränktheit* unterliegt.
Zur Vorbereitung einer neuen Interpolationsfunktion betrachte die rotationssymmetrische Funktion

$$F(x) = e^{-\|x\|^2/2}, \qquad F \in \mathcal{S}(\mathbb{R}^2)$$

F ist Fixpunkt der Fouriertransformation [16], d.h. es gilt $F = \hat{F}$.
F bzw. $\hat{F}$ ist natürlich nicht bandbeschränkt, aufgrund des exponentiellen Abfalls stellt dies aber kein wesentliches Problem dar (Abb. 5.3).
Die neue Interpolationsfunktion lautet :

$$\hat{w}(\xi) = e^{-\|R\,\xi\|^2/2} = \hat{F}_R$$

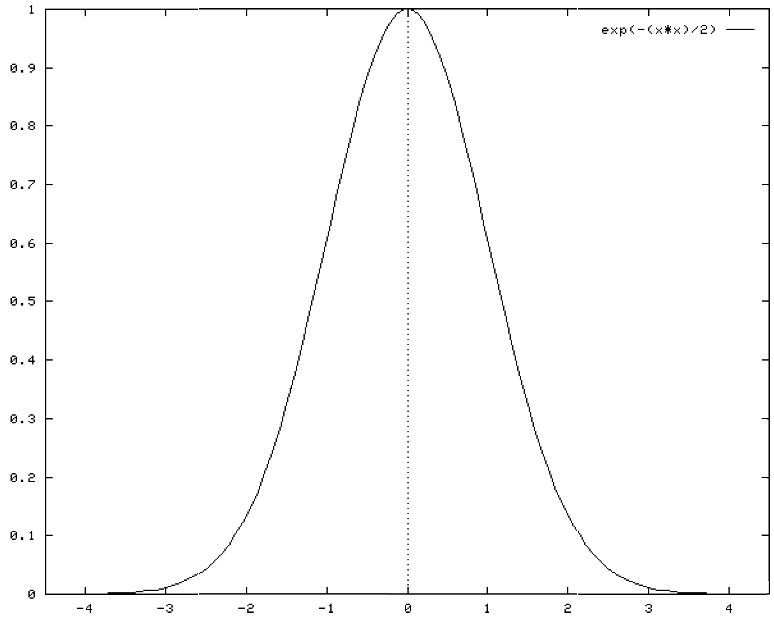


Abbildung 5.3: $\exp(-x^2/2) \qquad x \in R$

Ihre Form wird durch R in offensichtlicher Weise beeinflußt. Aus den Rechenregeln für die Fouriertransformation folgt :

$$w(x) = \frac{1}{R^2}\, e^{-\|\frac{1}{R}x\|^2/2} = \frac{1}{R^2}\, F_{\frac{1}{R}}$$

Man rekonstruiert nun nicht mehr f, sondern eine Funktion $u = f \cdot w$.
Die Fouriertransformierte $\hat{u} = (2\pi)^{-1}\hat{w} * \hat{f}$ ist wie $\hat{f}$ eine durch 1 bandbeschränkte Funktion, es werden also wieder Fourierwerte auf dem Gitter $\pi\mathbb{Z}^2$ gesucht. f kann in einem letzten Schritt in einfacher Weise aus u zurückgewonnen werden.
Geschwindigkeit und Rekonstruktionsqualität des modifizierten Verfahrens werden im wesentlichen durch folgende 3 Parameter gesteuert :
1. Die radiale Schrittweite im *Frequenzraum*, S.
2. Die Anzahl der im 2. Schritt miteinbezogenen kartesischen Nachbarn L.
3. Die Stauchung bzw. Streckung der Interpolationsfunktion durch R.

Ein modifizierter UFR–Algorithmus

Das Steuertripel lautet (L=25, S=$\pi/4$, R=0.55). Die radiale Schrittweite im *Frequenzraum* wird also auf $\pi/4$ herabgesetzt.

1. Schritt:

$$\hat{g}_{jr} = (2\pi)^{-\frac{1}{2}} \quad \frac{1}{q} \sum_{l=-q}^{q-1} e^{-i\pi lr/(4q)} g(\theta_j, s_l)$$

$\theta_j = (\cos\frac{j\pi}{p}, \sin\frac{j\pi}{p})$; $s_l = \frac{l}{q}$; $r = -4q, \ldots, 4q-1$; $j = 0, \ldots, p-1$.

Dies kann folgendermaßen auf vier FFT der Länge $2q$ reduziert werden:
Setze $\hat{h}_r := \hat{g}_{jr}, \quad h_l := g(\theta_j, s_l)$:

$$\begin{aligned}
\hat{h}_{4r} &= (2\pi)^{-\frac{1}{2}} \quad \frac{1}{q} \sum_{l=-q}^{q-1} e^{-i\pi lr/q}\, h_l \\
\hat{h}_{4r+1} &= (2\pi)^{-\frac{1}{2}} \quad \frac{1}{q} \sum_{l=-q}^{q-1} e^{-i\pi lr/q}\, e^{-i\pi l/(4q)}\, h_l \\
\hat{h}_{4r+2} &= (2\pi)^{-\frac{1}{2}} \quad \frac{1}{q} \sum_{l=-q}^{q-1} e^{-i\pi lr/q}\, e^{-i\pi l/(2q)}\, h_l \\
\hat{h}_{4r+3} &= (2\pi)^{-\frac{1}{2}} \quad \frac{1}{q} \sum_{l=-q}^{q-1} e^{-i\pi lr/q}\, e^{-i\pi(3l)/(4q)}\, h_l
\end{aligned}$$

$r = -q, \ldots, q-1$.
Diese Rechnung wird für alle Richtungen j durchgeführt, und man erhält die Fourierwerte auf einem Polargitter mit radialer Schrittweite $\pi/4$.

2. Schritt: Das Verfahren erlaubt die Beschränkung auf die 25 kartesischen Nachbarn des jeweiligen polaren Punktes ($L = 25$). Die vollständige Integrationsformel lautet:

$$\hat{u}_{k_1k_2} = (2\pi)^{-\frac{3}{2}} \sum_{j=0}^{p-1} \sum_{r=-4q}^{4q-1} \hat{w}\left(\pi(k_1, k_2) - r\frac{\pi}{4}(cos\frac{j\pi}{p}, sin\frac{j\pi}{p})\right)\ \hat{g}_{jr}\ J(j,r) \quad (5.5)$$

$k_1, k_2 = -q, \ldots, q-1, \qquad \hat{w} = \hat{F}_{0.55}$
Mit dem Skalierungsfaktor:

$$J(j,r) = \begin{cases} |r| \cdot \frac{\pi}{4}(\frac{\pi}{4}\frac{\pi}{p}) & : \quad r \neq 0 \\ 0.104375\ \pi & : \quad r = 0 \end{cases}$$

$J(j,r)$ muß in obiger Weise abgeändert werden, damit dem Nullpunkt des Polargitters nicht das Gewicht 0 zugeordnet wird. Andernfalls werden die Funktionswerte der Rekonstruktion vermindert und die Dichteunterschiede etwas verfälscht.
Wie bereits beschrieben, wird für jeden polaren Punkt das Produkt aus (5.5)

zum bisherigen Wert seiner L kartesischen Nachbarn addiert.
Die Reellwertigkeit von f führt zur Reellwertigkeit von u. Man beachte wieder, daß in diesem Fall nur die $\hat{u}_{k_1 k_2}$ der oberen Halbebene bestimmt werden müssen (siehe auch Abschnitt 2.1).

3. Schritt: Die Filterung im *Frequenzraum* ist auch bei diesem Verfahren unverzichtbar :

$$\hat{u}'_{k_1 k_2} = \begin{cases} \cos(\frac{\pi\|k\|}{2q})\,\hat{u}_{k_1 k_2} & : \quad \|k\| < q \\ 0 & : \quad \|k\| \geq q \end{cases}$$

$k = (k_1, k_2); \quad k_1, k_2 = -q, \ldots, q-1.$
Die Rücktransformation liefert u auf dem Gitter $\frac{1}{q}\mathbb{Z}^2$:

$$u_{mn} = \frac{\pi}{2} \sum_{k_1=-q}^{q-1} e^{i\pi m k_1/q} \sum_{k_2=-q}^{q-1} e^{i\pi n k_2/q}\, \hat{u}'_{k_1 k_2}$$

$m, n = -q, \ldots, q-1.$
Nun wird f durch die Gleichung $f = u/w$ zurückgewonnen ($R = 0.55$) :

$$f_{k_1 k_2} = R^2 e^{\frac{1}{2}(\|\frac{1}{R}k\|/q)^2} u_{k_1 k_2}$$

$k = (k_1, k_2); \quad k_1, k_2 = -q, \ldots, q-1.$

Die Bildtafeln im 6. Kapitel dokumentieren, daß mit dieser Technik ein Algorithmus konstruierbar ist, der mit einem Rechenaufwand der Ordnung $O(q^2 \log q)$ allen bisher beschriebenen Fourier–Verfahren überlegen ist. Die hier gewählte Kombination aus L, radialer Schrittweite und Interpolationsfunktion ist nur eine von vielen möglichen Lösungen und sicher nicht optimal. Der modifizierte UFR-Algorithmus erreicht eine der gefilterten Rückprojektion gleichwertige Rekonstruktionsqualität (vergleiche Abb. 6.13c und 6.13d). Eine weitergehende Untersuchung wird in dieser Arbeit nicht durchgeführt. In den numerischen Experimenten findet sich eine Version mit dem Steuertripel (L=9, S=$\pi/3$, R=1.0). Sie erreicht nicht die Rekonstruktionsqualität der obigen Version. Es soll jedoch ausdrücklich darauf hingewiesen werden, daß das Verfahren mit diesen Parametern den Algorithmus von Löw-Natterer in der Ausführungsgeschwindigkeit und in der Rekonstruktionsgüte übertrifft. Der UFR–Algorithmus zeigt sich als das vielversprechendste Verfahren zur Fourierrekonstruktion.

Kapitel 6

Numerische Experimente

6.1 Praktische Erfahrungen

Zusätzlich zu den Verfahren aus Kapitel 2 wurde eine modifizierte Version der bilinearen Methode getestet. Wie beim Verfahren von Löw-Natterer wurde im 1. Schritt die radiale Schrittweite im Frequenzraum halbiert, d.h. auf $\pi/2$ gesetzt. Trotz der Verbesserung der Rekonstruktion bleibt das Grundproblem der Methode – wie in Abschnitt 2.2 dargestellt – erhalten. Die Rekonstruktionsergebnisse sind ebenfalls unbefriedigend, nur das Objekt 1 (siehe Abb. 6.7d) wurde wiedererkennbar rekonstruiert.

Das Verfahren von Pasciak neigt dazu, auf den Diagonalen des rekonstruierten Bildes streifenförmige Schatten zu erzeugen. Dieser Fehler wird stärker, wenn das Verhältnis $p \geq \pi q$ nicht eingehalten wird oder p nicht durch 4 teilbar ist. In der Nähe der Diagonalen ist die Art der Interpolation im 2. Schritt kritisch. Die Abstände in radialer Richtung zu den kartesischen Punkten nehmen dort zu (siehe auch den Beweis zu Satz 3.1). Dieser charakteristische Fehler wird schwächer, wenn die Anzahl der Richtungen über das Standardverhältnis hinaus erhöht wird. Insgesamt erzeugt der Algorithmus brauchbare Bilder, er erreicht aber nicht die Qualität einer gut implementierten gefilterten Rückprojektion.

Der Algorithmus von Löw-Natterer erzeugt bessere Rekonstruktionen, der Rechenaufwand ist allerdings höher. Eine interessante Eigenschaft dieses Verfahrens ist seine Stabilität bzgl. der Herabsetzung der Richtungszahl p unter das Standardverhältnis $p \geq \pi q$. Selbst bei $p = 2q$ erreichte es immer noch eine akzeptabele Bildqualität. Die Rekonstruktion mit dem Verfahren von Pasciak ist bei diesem Verhältnis von einem intensiven, diagonal gerichteten Störmuster überzogen. Die Art der Interpolation im Verfahren von

Löw-Natterer, die um Exaktheit in radialer Richtung bemüht ist, ist dem Problem gut angepaßt. Der Wert für die Abbruchkonstante L wurde experimentell bestimmt und sollte für Abtastgeometrien mit $q \leq 320$ ausreichen. Das Verfahren ist der gefilterten Rückprojektion im niederfrequenten Bereich unterlegen, in der Hochkontrastauflösung dagegen leicht überlegen [4]. Die Bildfehler lassen sich als fleckig, im Falle der Rückprojektion dagegen als wirbel- bzw. streifenförmig charakterisieren.
Die Methode von Stark ist in ihrer ursprünglichen Form [5] mit hohem Rechenaufwand verbunden. Das Verfahren von Löw-Natterer kann durch eine Interpolation höherer Ordnung in azimutaler Richtung etwas verbessert werden, sodaß eine Rekonstruktionsqualität wie bei konsequenter Durchführung der Idee von Stark möglich ist (siehe Abb. 6.12d).
Wie in den Bildtafeln erkennbar, ist der Qualitätssprung von der bilinearen Methode zu den Verfahren aus den Kapiteln 3 und 4 sehr groß. Die lineare Interpolation im 2. Schritt ist in diesen Verfahren obligatorisch und nicht mit wesentlich höherem Rechenaufwand verbunden. Allerdings treten die Qualitätsunterschiede der Verfahren von Pasciak, Löw-Natterer und Stark nur bei sehr genauer Messung der Durchstrahlungsdaten hervor. In den hier durchgeführten Experimenten wurde jedes Linienintegral des Objekts aus 21 Messungen über die Detektorbreite $1/q$ gemittelt. Dies kann auch als Simulation der Bandbeschränktheit des Testobjekts angesehen werden. Wird dies auf 1 bis 3 Messungen reduziert, ist der Mehraufwand des Verfahrens von Löw-Natterer gegenüber dem Verfahren von Pasciak nicht lohnend. Die harten Kanten der Testobjekte schlagen sich in den simulierten Durchstrahlungsdaten nieder. Ein Test mit realistischen – vielleicht leichter zu rekonstruierenden – Objekten z. B . aus der Medizin wäre daher interessant . In [4] ist von Artefakten die Rede, " die in praxi wegen stärkerer Bandbegrenzung der Objekte nicht so bedeutsam sind."

Mit dem unter dem Namen UFR-Algorithmus vorgestellten Verfahren wurden hervorragende Resultate erzielt. Die Methode ist hinsichtlich der Rekonstruktionsqualität den anderen Fourier–Algorithmen überlegen. Sie reagiert bei einem ungenauen Datensatz deutlich stabiler. Die aufwendigste der hier getesteten Versionen, im 5. Kapitel als modifizierter UFR-Algorithmus vorgestellt (L=25, S=$\pi/4$, R=0.55), liefert auch noch bei einer Richtungszahl von $p = 2q$ gute Ergebnisse. Allerdings wurden hier nicht alle Fragen zum UFR–Algorithmus geklärt. Weiterführende Experimente mit anderen Interpolationsfunktionen könnten den Rechenaufwand herabsetzen, ohne die hier schon erreichte Rekonstruktionsqualität zu vermindern. Die Methode stellt eine ernstzunehmende Konkurrenz zur gefilterten Rückprojektion dar.
Für die erstellten Programme wurden, soweit möglich, ähnliche Prozedu-

ren verwendet, die nicht bis ins kleinste Detail optimiert wurden (FFT mit floating-point-Arithmetik ohne spezielle Routinen für die Behandlung reeller Vektoren, etc.). Abbildung 6.1 gibt die Ausführungszeiten (Sekunden) einer Rekonstruktion mit $p = 400$, $q = 128$ auf verschiedenen Rechnern an.

Programm	PC [1]	SUN 4	SUN 10
Bilineare Methode	12	8	4
Pasciak	38	24	13
Löw-Natterer (L=12)	54	44	23
L-N (L=23)	67	57	29
L-N (L=30)	75	68	35
L-N mit vollst. Sincreihe [2]	131	104	46
UFR (L=9, S=$\pi/2$, R=0.75)	38	23	12
UFR (L=9, S=$\pi/3$, R=1.0)	51	32	16
UFR (L=25, S=$\pi/4$, R=0.55)	137	90	43
Gef. Rückprojektion		176	102

Abbildung 6.1: *Ausführungszeiten der Programme in Sekunden*

6.2 Bildmaterial

Zu den Rekonstruktionen ist folgendes zu sagen :
Objekt 1 ist ein einfach zu rekonstruierendes Gebilde, welches hauptsächlich zum Test der schwächeren Verfahren dient. Der Rand des Objekts hat den Funktionswert 90, die inneren Ellipsen setzen sich durch verhältnismäßig große Dichteunterschiede vom Hintergrund ab.
Objekt 2 ist anspruchsvoller und soll die Leistungsfähigkeit der hier vorgestellten Versionen der Algorithmen der Kapitel 3 bis 5 demonstrieren. Der Rand des Objekts hat den Funktionswert 200, die inneren Ellipsen unterscheiden sich nur um den Wert 1 bzw. 2 vom inneren Hintergrund.
Objekt 3 dient zum Vergleich der Verfahren aus den Kapiteln 4 und 5. Hier interessiert vor allem die Homogenität des Hintergrundes.

[1] PC AT-Kompatibel, 50MHz, 32-Bit Betriebssystem
SUN-Workstations mit RISC-Prozessoren

[2] Ohne Überabtastung im ersten Schritt

Bei allen Rekonstruktionen wurde der $cos-$Filter benutzt. Die folgenden Tabellen geben Fehlergrößen zwischen dem – ungefilterten – Original und den jeweiligen Rekonstruktionen an. Relativer L_1-Fehler und relativer L_2-Fehler sind einzeln betrachtet wenig aussagekräftig [9]. Letztlich sind die Bildtafeln am besten geeignet, die Rekonstruktionsgüte zu beurteilen.

Figur [1]	Mx	My	Rx	Ry	Wx	F (Obj. 1)	F (Obj. 2)
Kreis	0.0	0.105	0.055			10	1
Kreis	0.0	-0.6	0.025			10	1
Ellipse	-0.015	-0.1	0.07	0.052	0.0	20	2
Ellipse	0.0	0.0	0.7	0.9	0.0	90	200
Ellipse	0.0	-0.018	0.67	0.85	0.0	-40	-98
Ellipse	0.06	-0.6	0.025	0.05	0.0	10	1
Ellipse	-0.08	-0.6	0.05	0.025	0.0	10	1
Ellipse	0.0	0.35	0.23	0.25	0.0	10	1
Ellipse	0.25	0.0	0.13	0.31	-0.32	-20	-2
Ellipse	-0.25	0.01	0.18	0.42	0.32	-20	-2

Abbildung 6.2: Objekt 1, Objekt 2

[1](Mx, My) ist Mittelpunkt der Figur, F ist Funktionswert. Kreis: Rx ist Radius. Ellipse: Rx, Ry sind die Halbachsen, Wx ist Winkel der ersten Halbachse mit der x-Achse. Rechteck: Rx, Ry sind die halben Seitenlängen.

Figur	Mx	My	Rx	Ry	Wx	F
Kreis	-0.6	-0.35	0.05			4
Kreis	-0.3	-0.35	0.05			8
Kreis	0.0	-0.35	0.05			12
Kreis	0.3	-0.35	0.05			16
Kreis	0.6	-0.35	0.05			20
Kreis	0.0	0.9	0.05			-2000
Kreis	0.0	0.9	0.1			1000
Ellipse	0.0	0.0	0.8	1.0	0.0	1000
Rechteck	-0.6	0.25	0.006	0.2		-25
Rechteck	-0.45	0.25	0.007	0.2		-25
Rechteck	-0.3	0.25	0.008	0.2		-25
Rechteck	-0.15	0.25	0.009	0.2		-25
Rechteck	0.0	0.25	0.01	0.2		-25
Rechteck	0.15	0.25	0.0125	0.2		-25
Rechteck	0.3	0.25	0.015	0.2		-25
Rechteck	0.45	0.25	0.0175	0.2		-25
Rechteck	0.6	0.25	0.02	0.2		-25

Abbildung 6.3: Objekt 3

Algorithmus	$rel. L_1 - Fehler$	$rel. L_2 - Fehler$
Nearest-Neighbourhood	47.64 %	44.54 %
Bilineare Methode	26.42 %	27.30 %
Bilin. Meth. mit Überabtast.	8.19 %	12.67 %
Pasciak ohne lin. Interp.	4.72 %	10.63 %
L-N ohne lin. Interp. (L=23)	3.63 %	10.42 %
Pasciak	3.41 %	10.51 %
L-N (L=23)	3.30 %	10.40 %
UFR (L=9, S=$\pi/2$, R=0.75)	3.44 %	10.40 %
UFR (L=9, S=$\pi/3$, R=1.0)	3.30 %	10.39 %
UFR (L=25, S=$\pi/4$, R=0.55)	3.27 %	10.39 %

Abbildung 6.4: rel. Rekonstruktionsfehler, Objekt 1, p=400, q=128

Algorithmus	*rel.* $L_1 - Fehler$	*rel.* $L_2 - Fehler$
Bilineare Methode	25.65 %	27.52 %
Bilin. Meth. mit Überabtast.	7.91 %	13.04 %
Pasciak ohne lin. Interp.	4.61 %	10.97 %
L-N ohne lin. Interp. (L=23)	3.42 %	10.74 %
Pasciak	3.19 %	10.87 %
L-N (L=23)	3.06 %	10.73 %
UFR (L=9, S=π/3, R=1.0)	3.06 %	10.71 %
UFR (L=25, S=π/4, R=0.55)	3.02 %	10.72 %

Abbildung 6.5: rel. Rekonstruktionsfehler, Objekt 2, p=400, q=128

Algorithmus	*rel.* $L_1 - Fehler$	*rel.* $L_2 - Fehler$
Pasciak	2.06 %	7.60 %
L-N (L=23)	1.96 %	7.50 %
mod. L-N (L=23)	1.93 %	7.55 %
Stark (siehe Abschnitt 4.2)	1.92 %	7.53 %
UFR (L=9, S=π/3, R=1.0)	1.92 %	7.44 %
UFR (L=25, S=π/4, R=0.55)	1.87 %	7.45 %

Abbildung 6.6: rel. Rekonstruktionsfehler, Objekt 3, p=256, q=80