

Diskrete Kosinustransformation in der Bildverarbeitung

Julius Berner, 1204596

01.07.2016

Inhaltsverzeichnis

1	Diskrete Fouriertransformation	3
1.1	Motivation durch Fourierreihen	3
1.2	Herleitung der DFT	4
1.3	Eigenschaften der DFT	8
2	Diskrete Kosinustransformation	9
2.1	Herleitung der DCT II	10
2.2	Übersicht weiterer Kosinustransformationen	13
2.3	DCT II in der Bildverarbeitung	17
2.3.1	Kodierung	18
2.3.2	Dekodierung	21
3	Shannon Sampling Theorem	23
3.1	Grundlagen der Fouriertransformation	23
3.2	Theorem und Beweis	25

Zusammenfassung

Die vorliegende Arbeit soll eine Einführung zur diskreten Kosinustransformation und deren Verwendung in der Bildverarbeitung sein. Nachdem einleitend mit der diskreten Fouriertransformation begonnen wird, folgt ein Überblick über die acht diskreten Kosinustransformationen. Die Rolle der diskreten Kosinustransformation II in der effizienten Komprimierung von Bildern wird dabei genauer betrachtet. Abschließend wird auf den Übergang von einem kontinuierlichen zu einem diskreten Signal in Form des Shannon Sampling Theorems eingegangen.

1 Diskrete Fouriertransformation

Die diskrete Fouriertransformation (DFT) ist für die Transformation und Analyse von Vektoren (diskreten Signalen) ebenso grundlegend wie die Fourierreihe bei periodischen Funktionen (kontinuierlichen Signalen). Als Motivation sollen deshalb grundlegende Resultate zu Fourierreihen wiederholt werden.

1.1 Motivation durch Fourierreihen

Sei $L > 0$, dann betrachten wir im Folgenden den Hilbertraum $L_2(0, L)$ der (bezüglich dem Lebesgue-Maß) quadratintegrierbaren Funktionen auf $(0, L)$ mit Skalarprodukt $\langle u, v \rangle = \frac{1}{L} \int_0^L u(t)\overline{v(t)}dt$ und induzierter Norm $\|u\|_2$.

Satz 1.1. Die Funktionen $\{e_k(t) = e^{\frac{2\pi ikt}{L}}, k \in \mathbb{Z}\}$ bilden eine Orthonormalbasis von $L_2(0, L)$.

Beweis. in [1]. □

Bemerkung 1.2.

1. Dies ist äquivalent dazu, dass jede Funktion $u \in L_2(0, L)$ in eine (in L_2) konvergente Fourierreihe entwickelt werden kann

$$u(t) \stackrel{L_2}{=} \sum_{k \in \mathbb{Z}} c_k e^{\frac{2\pi ikt}{L}} \quad (1)$$

mit

$$c_k = \frac{1}{L} \int_0^L u(t) e^{-\frac{2\pi ikt}{L}} dt$$

(die Konvergenz der Reihe ist hierbei im quadratischen Mittel bzw. in der L_2 -Norm zu verstehen, d.h. $\lim_{n \rightarrow \infty} \|u - \sum_{k=-n}^n c_k e_k\|_2 = 0$)

2. Bezüglich der Konvergenz der Fourierreihe gibt es eine Vielzahl an weiteren Resultaten (siehe [3],[6]). Beispielsweise gilt auch gleichmäßige Konvergenz der Reihe in (1), falls u stetig differenzierbar ist.
3. Da die Fourierreihe L -periodisch ist, geht man oft davon aus, dass u L -periodisch auf ganz \mathbb{R} fortgesetzt ist.

Die Darstellung in (1) kann als Zerlegung der Funktion u in ihre Frequenzen aufgefasst werden. Die Funktion $c_k e_k(t)$ ist periodisch mit Frequenz $\frac{k}{L}$, Amplitude $|c_k e_k(t)| = |c_k|$ und (Null-)Phasenwinkel $\varphi_k = \arg(c_k e_k(0)) = \arg(c_k)$.

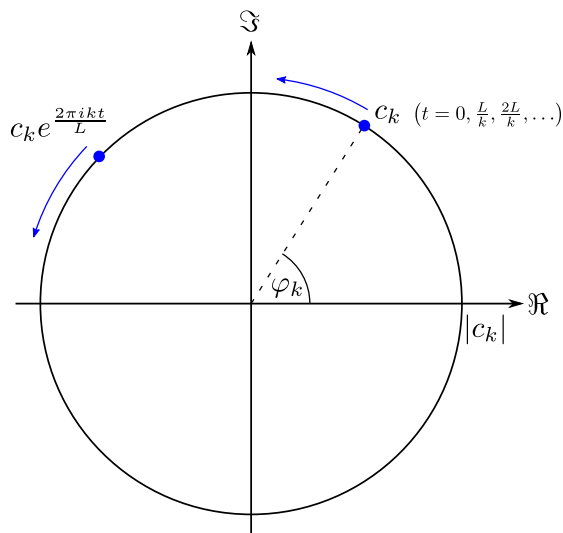


Abbildung 1: komplexe Exponentialfunktion $t \rightarrow c_k e^{\frac{2\pi i k t}{L}}$

Somit gibt der Koeffizient c_k den Anteil der Frequenz $\frac{k}{L}$ innerhalb der Funktion u an und die Fourierreihenentwicklung bildet ein kontinuierliches Signal $u(t)$ auf ein diskretes Frequenzspektrum $\{c_k\}_{k \in \mathbb{Z}}$ ab. Wenn man die Variable t als Zeit interpretiert, spricht man oft davon, dass die Fourierreihenentwicklung den kontinuierlichen Zeitbereich, auf einen diskreten Frequenzbereich abbildet.

Allgemein ist es für viele Anwendungen in der Praxis, aber auch als mathematisches Hilfsmittel, von Vorteil Funktionen bzw. Signale in ihre Frequenzen zu zerlegen. Dies kann man folgendermaßen gliedern:

	kontinuierliche Frequenzen	diskrete Frequenzen
diskretes Signal	Fouriertransformation für zeitdiskrete Signale (DTFT)	Diskrete Fouriertransf./ Kosinustranf. (DFT/DCT)
kontinuierliches Signal	Fouriertransformation (FT)	Fourierreihe

Dabei wird in dieser Arbeit auf alle oben genannten Transformationen außer der DTFT eingegangen.

1.2 Herleitung der DFT

Wir werden nun nach [3] die diskrete Fouriertransformation als Lösung eines Interpolationsproblems herleiten. Wie bei der Fourierreihe gehen wir dabei von einer L -periodischen Funktion u aus, von der wir aber nur noch N äquidistante Funktionswerte y_j innerhalb einer Periode kennen:

$$y_j = u\left(\frac{jL}{N}\right) \quad \text{für } j = 0, \dots, N-1$$

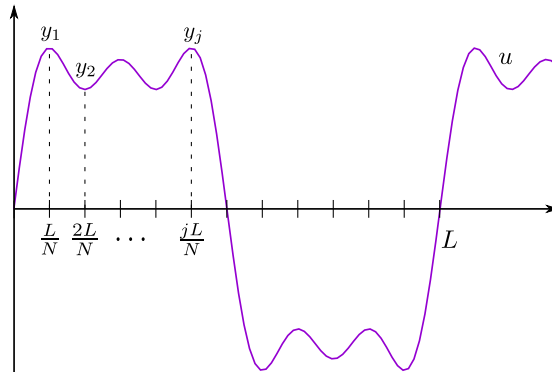


Abbildung 2: Abtasten der Funktion u

Man spricht davon, dass die Funktion bzw. das Signal alle $\frac{L}{N}$ Zeiteinheiten *abgetastet* wurde (oft wird auch der englische Begriff *Sampling* für den Prozess des Abtastens verwendet). Der Übergang von einer kontinuierlichen Funktion zu diskreten Werten wird in Kapitel 3 noch genauer betrachtet.

Ausgehend von diesen N Funktionswerten wollen wir N Fourierkoeffizienten c_k von u approximieren. Da man zeigen kann, dass die Fourierkoeffizienten für $|k| \rightarrow \infty$ gegen 0 streben, versuchen wir c_k für $k = -\frac{N}{2}, \dots, \frac{N}{2} - 1$ (oder falls N ungerade $k = -\frac{N-1}{2}, \dots, \frac{N-1}{2}$) anzunähern. Sei im Folgenden N o.B.d.A. gerade. Dann wollen wir die Koeffizienten \tilde{c}_k des trigonometrischen Polynoms

$$p(t) = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \tilde{c}_k e^{\frac{2\pi i k t}{L}} \quad (2)$$

berechnen, welches u an den Stellen $\frac{jL}{N}$ ($j = 0, \dots, N-1$) interpoliert. Motiviert durch die Darstellung in (1) hoffen wir, dass die \tilde{c}_k eine gute Approximation der Fourierkoeffizienten darstellen. In Lemma 1.5 werden wir später den Fehler $|c_k - \tilde{c}_k|$ genauer untersuchen.

Nach Einsetzen der Interpolationsbedingungen erhalten wir

$$y_j = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \tilde{c}_k e^{\frac{2\pi i j k}{N}} = \sum_{k=0}^{N-1} \hat{y}_k e^{\frac{2\pi i j k}{N}} \quad \text{für } j = 0, \dots, N-1$$

wobei die Periodizität der komplexen Exponentialfunktion ausgenützt wurde und

$$\hat{y}_k := \begin{cases} \tilde{c}_k, & k = 0, \dots, \frac{N}{2} - 1 \\ \tilde{c}_{k-N}, & k = \frac{N}{2}, \dots, N-1 \end{cases} \quad (3)$$

um ein vorteilhaftes Summationsintervall zu erhalten. Wir können dies vektori-

siert schreiben als

$$\underbrace{\begin{pmatrix} y_1 \\ \vdots \\ y_j \\ \vdots \\ y_{N-1} \end{pmatrix}}_y = \widehat{y}_0 \underbrace{\begin{pmatrix} 1 \\ \vdots \\ 1 \\ \vdots \\ 1 \end{pmatrix}}_{v_0} + \dots + \widehat{y}_k \underbrace{\begin{pmatrix} 1 \\ \vdots \\ e^{\frac{2\pi ijk}{N}} \\ \vdots \\ e^{\frac{2\pi i(N-1)k}{N}} \end{pmatrix}}_{v_k} \dots + \widehat{y}_{N-1} \underbrace{\begin{pmatrix} 1 \\ \vdots \\ e^{\frac{2\pi ij(N-1)}{N}} \\ \vdots \\ e^{\frac{2\pi i(N-1)(N-1)}{N}} \end{pmatrix}}_{v_{N-1}} \quad (4)$$

Lemma 1.3. Die Vektoren $\{v_k, k = 0, \dots, N-1\}$ sind orthogonal mit Norm \sqrt{N} bezüglich des Standard-Skalarproduktes auf dem \mathbb{C}^N .

Beweis. Mit der Summenformel für die geometrischen Reihe ergibt sich:

$$\langle v_k, v_l \rangle = \sum_{j=0}^{N-1} e^{\frac{2\pi ijk}{N}} e^{-\frac{2\pi ijl}{N}} = \sum_{j=0}^{N-1} \left(e^{\frac{2\pi i(k-l)j}{N}} \right)^j = \begin{cases} N, & k = l \\ 0, & \text{sonst} \end{cases}$$

□

Durch Anwenden des Skalarproduktes mit v_k im zweiten Argument auf (4), erhält man leicht die Formel für \widehat{y}_k :

$$\widehat{y}_k = \frac{1}{N} \langle y, v_k \rangle = \frac{1}{N} \sum_{j=0}^{N-1} y_j e^{-\frac{2\pi ijk}{N}} \quad k = 0, \dots, N-1 \quad (5)$$

und damit durch Umbenennung wie in (3) auch die Formel für die Koeffizienten \tilde{c}_k des Interpolationspolynoms. Aufgrund der Linearität kann man dies auch in Matrix-Schreibweise angeben:

$$\begin{pmatrix} \widehat{y}_1 \\ \vdots \\ \widehat{y}_k \\ \vdots \\ \widehat{y}_{N-1} \end{pmatrix} = \frac{1}{N} \underbrace{\begin{bmatrix} e^{-\frac{2\pi ijk}{N}} \end{bmatrix}_{k,j}}_{:=F} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_k \\ \vdots \\ y_{N-1} \end{pmatrix} = \frac{1}{N} \begin{pmatrix} \overline{v_0^\top} \\ \vdots \\ \overline{v_k^\top} \\ \vdots \\ \overline{v_{N-1}^\top} \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_k \\ \vdots \\ y_{N-1} \end{pmatrix} \quad (6)$$

Sei $\widehat{y} := (\widehat{y}_1, \dots, \widehat{y}_k, \dots, \widehat{y}_{N-1})$, dann nennt man die Abbildung

$$\mathcal{F}_N : \mathbb{C}^N \mapsto \mathbb{C}^N \\ y \mapsto \widehat{y} = \frac{1}{N} F \cdot y$$

diskrete Fouriertransformation (DFT).

Da die Vektoren v_k orthogonal mit Norm \sqrt{N} sind und die Fouriermatrix F symmetrisch ist, gilt für die inverse Matrix $F^{-1} = \frac{1}{N} \overline{F}$ (wobei wir \overline{F} für die komponentenweise Konjugierte schreiben). So ist die *inverse diskrete Fouriertransformation (iDFT)* gegeben durch:

$$\mathcal{F}_N^{-1} : \mathbb{C}^N \mapsto \mathbb{C}^N \\ \widehat{y} \mapsto y = \overline{F} \cdot \widehat{y}$$

bzw. komponentenweise

$$y_k = \langle \hat{y}, \bar{v}_k \rangle = \sum_{j=0}^{N-1} \hat{y}_j e^{\frac{2\pi i j k}{N}} \quad k = 0, \dots, N-1$$

Bemerkung 1.4 (Normierung). *Die Normierung $\frac{1}{N}$ ist in vielen Quellen bzw. Anwendungen verschieden. Oft wird die Transformation unitär gemacht, indem sowohl iDFT als auch DFT mit dem Faktor $\sqrt{\frac{1}{N}}$ multipliziert werden. Alternativ kann man anstatt der DFT auch die iDFT mit $\frac{1}{N}$ normieren.*

Erstaunlicherweise bekommt man dieselbe Formel für die Koeffizienten \tilde{c}_k , wenn man die Integralformeln für die echten Fourierreihenoeffizienten c_k mit der Trapezformel annähert (siehe [3]). Das nächste Lemma klärt, inwiefern dies eine gute Approximation darstellt.

Lemma 1.5 (Beziehung zwischen den exakten und den durch die DFT approximierten Fourierkoeffizienten). *Sei u eine L -periodische Funktion mit einer absolut konvergenten Fourierreihe ¹*

$$u(t) = \sum_{k \in \mathbb{Z}} c_k e^{\frac{2\pi i k t}{L}}$$

$$\sum_{k \in \mathbb{Z}} |c_k| < \infty$$

dann gilt

$$\tilde{c}_k - c_k = \sum_{q \neq 0} c_{k+qN} \quad (7)$$

wobei \tilde{c}_k die durch die DFT berechneten (und entsprechend umsortierten) Koeffizienten des trigonometrischen Interpolationspolynoms (2) vom Grad N sind.

Beweis. Wegen der Interpolationseigenschaft gilt:

$$u\left(\frac{jL}{N}\right) = y_j = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \tilde{c}_k e^{\frac{2\pi i j k}{N}}$$

Andererseits folgt aus der absoluten Konvergenz der Fourierreihe

$$\begin{aligned} u\left(\frac{jL}{N}\right) &= \sum_{k \in \mathbb{Z}} c_k e^{\frac{2\pi i j k}{N}} = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \sum_{q \in \mathbb{Z}} c_{k+qN} e^{\frac{2\pi i j (k+qN)}{N}} \\ &= \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \left(\sum_{q \in \mathbb{Z}} c_{k+qN} \right) e^{\frac{2\pi i j k}{N}} \end{aligned}$$

wobei wir zuerst die Terme mit einem fixen Index modulo N summieren und die Periodizität der Exponentialfunktion ausnützen. Insgesamt folgt durch Koeffizientenvergleich $\tilde{c}_k = \sum_{q \in \mathbb{Z}} c_{k+qN}$ und damit die Aussage. \square

¹gilt z.B. für u stetig differenzierbar, siehe Bemerkung 1.2

Umso schneller die Koeffizienten c_k gegen Null gehen, umso besser ist also die Approximation $\tilde{c}_k \approx c_k$, für $\frac{N}{2} \leq k \leq \frac{N}{2} - 1$. Aus den Eigenschaften der Fourierreihe (je glatter, desto schneller konvergieren die Koeffizienten gegen Null, siehe z.B. [3]) schließen wir, dass die Güte der Approximation also von der Glattheit der Funktion u abhängt.

Außerdem ist die Summe aus (7) endlich, falls u ein trigonometrisches Polynom ist, und wir können folgende Bemerkung ableiten.

Bemerkung 1.6. Falls u ein trigonometrisches Polynom vom Grad P ist, d.h. $u(t) = \sum_{k=-P}^P c_k e^{\frac{2\pi i k t}{L}}$, dann gilt $\tilde{c}_k = c_k$, falls $N \geq 2P + 1$. In Worten: Die approximierten Fourierkoeffizienten sind exakt, falls die Anzahl der Stützstellen bzw. die Dimension der DFT mehr als doppelt so groß wie der Grad ist.

1.3 Eigenschaften der DFT

1. Parseval Identität

Seien $y, z \in \mathbb{C}^N$ und \hat{y}, \hat{z} die jeweiligen Bilder der N-dimensionalen diskreten Fouriertransformation, dann gilt:

$$\langle y, z \rangle = N \langle \hat{y}, \hat{z} \rangle$$

Daraus folgt unmittelbar für $z = y$:

$$\|y\| = \sqrt{N} \|\hat{y}\|$$

Beweis. Mit der Darstellung aus (4) gilt:

$$\begin{aligned} \langle y, z \rangle &= \left\langle \sum_{k=0}^{N-1} \hat{y}_k v_k, \sum_{l=0}^{N-1} \hat{z}_l v_l \right\rangle \\ &= N \sum_{k=0}^{N-1} \hat{y}_k \overline{\hat{z}_k} = N \langle \hat{y}, \hat{z} \rangle \end{aligned}$$

□

2. Periodizität

Für die folgende Eigenschaft (aber auch allgemein) bietet es sich an, den Vektor $y = (y_0, \dots, y_{N-1})$ N-periodisch zu einer Folge in $\mathbb{C}^{\mathbb{N}}$ fortzusetzen. Dies entspricht auch der ursprünglichen Definition $y_k = u\left(\frac{kL}{N}\right)$, da u Periode L besitzt. Wir legen also fest:

$$y_k = u\left(\frac{kL}{N}\right) = u\left(\frac{kL}{N} + L\right) = u\left(\frac{(k+N)L}{N}\right) =: y_{k+N}$$

Aus der Definition (5) ergibt sich, dass auch die \hat{y}_k N-periodisch sind.

$$\hat{y}_k = \sum_{j=0}^{N-1} y_j e^{\frac{2\pi i j k}{N}} = \sum_{j=0}^{N-1} y_j e^{\frac{2\pi i j (k+N)}{N}} =: \hat{y}_{k+N}$$

3. Faltung \leftrightarrow Multiplikation

Seien $x, y \in \mathbb{C}^N$ zwei komplexe N -periodische Folgen und \hat{y}, \hat{z} die jeweiligen Bilder der N -dimensionalen diskreten Fouriertransformation im obigen Sinn. Wir definieren die diskrete, periodische Faltung $y * z$ von y und z als Folge $w \in \mathbb{C}^N$ mit

$$w_j := \sum_{l=0}^{N-1} y_l z_{j-l}, \quad j \in \mathbb{Z}$$

und die punktweise Multiplikation $y \times z$ als Folge $u \in \mathbb{C}^N$ mit

$$u_k := y_k z_k, \quad k \in \mathbb{Z}$$

Dann gilt, dass die DFT die Faltung in eine Multiplikation überführt und umgekehrt:

$$\begin{aligned} y * z &\xrightarrow{\mathcal{F}_N} N \hat{y} \times \hat{z} \\ y \times z &\xrightarrow{\mathcal{F}_N} \hat{y} * \hat{z} \end{aligned}$$

Beweis. Aus der Definition folgt

$$(\widehat{y * z})_k = \frac{1}{N} \sum_{j=0}^{N-1} \sum_{l=0}^{N-1} y_l z_{j-l} e^{-\frac{2\pi i k j}{N}}$$

und Vertauschen der Summen ergibt

$$(\widehat{y * z})_k = \frac{1}{N} \sum_{l=0}^{N-1} y_l e^{-\frac{2\pi i k l}{N}} \sum_{j=0}^{N-1} z_{j-l} e^{-\frac{2\pi i k (j-l)}{N}} = N \hat{y}_k \hat{z}_k$$

Die umgekehrte Relation folgt analog mit der iDFT. \square

Interessanterweise gilt ein ähnliches Resultat auch im Fall der Fouriertransformation und -reihen (siehe [3]).

4. Fast Fourier Transform (FFT)

Die Berühmtheit und weite Verbreitung der diskreten Fouriertransformation beruht zum großen Teil auf einem Algorithmus von J.W. Cooley und J.W. Turkey aus dem Jahr 1965, der die $O(N^2)$ Rechenoperationen, welche man zur naiven Berechnung von \hat{y} mittels des Matrix-Vektor Produktes in (6) benötigt, auf $O(N \log(N))$ reduziert. Dieser wird Fast Fourier Transform genannt und verwendet die besondere Gestalt der Fouriermatrix F um Rechenoperationen einzusparen.

2 Diskrete Kosinustransformation

Während in der Praxis viele Signale reellwertig sind, ist die diskrete Fouriertransformation eine komplexwertige Transformation und dementsprechend sind die Bilder eines reellen Signals im Allgemeinen komplex. Die Suche nach einer reellwertigen diskreten Transformation werden wir wieder mit dem Analogon der Fourierreihe motivieren.

2.1 Herleitung der DCT II

Sei eine gerade reelle Funktion auf $(-L, L)$ gegeben, d.h. $u(t) = u(-t)$. Nehmen wir an die Fourierreihe konvergiere absolut (z.B. u stetig differenzierbar, siehe Bemerkung 1.2), dann bleiben nur die Kosinus-Terme in der Reihenentwicklung erhalten, da:

$$\begin{aligned} c_k &= \frac{1}{2L} \int_{-L}^L u(t) e^{-\frac{\pi i k t}{L}} dt = \frac{1}{2L} \int_{-L}^L u(-t) e^{\frac{\pi i k t}{L}} dt \\ &= \frac{1}{2L} \int_{-L}^L u(t) e^{\frac{\pi i k t}{L}} dt = c_{-k} \end{aligned}$$

Nachdem bei einer reellen Funktion per Definition $\bar{c}_k = c_{-k}$ gilt, folgt $c_k = c_{-k} \in \mathbb{R}$ und

$$c_k e^{\frac{\pi i k t}{L}} + c_{-k} e^{-\frac{\pi i k t}{L}} = 2c_k \operatorname{Re}(e^{\frac{\pi i k t}{L}}) = 2c_k \cos \frac{\pi k t}{L}$$

Wenden wir dies auf die jeweiligen Paare in der Fourierreihenentwicklung an, dann erhalten wir mit $a_k := 2c_k$, $k \in \mathbb{Z}$

$$u(t) = \frac{a_0}{2} + \sum_{k=1}^{\infty} a_k \cos \frac{k\pi t}{L} \quad (8)$$

Analog zur Einführung der diskreten Fouriertransformation wollen wir bei der diskreten Kosinustransformation nun von einer abgetasteten geraden Funktion auf $(-L, L)$ ausgehen und als Interpolationspolynom eine Partialsumme der Reihe in (8) benützen. Die Funktion soll an den Zeitpunkten $\frac{(j+0.5)L}{N}$ ($j = 0, \dots, N-1$) abgetastet und interpoliert werden.

$$y_j = u\left(\frac{(j+0.5)L}{N}\right)$$

Die willkürlich erscheinende Verschiebung um 0.5 wird später motiviert und Alternativen aufgezeigt.

Wir wollen diesmal (motiviert durch die Darstellung in (8)) die Koeffizienten \hat{a}_k des (geraden) trigonometrischen Polynoms

$$p(t) = \frac{\hat{a}_0}{2} + \sum_{k=1}^{N-1} \hat{a}_k \cos \frac{k\pi t}{L}$$

berechnen, welches u an den Stellen $\frac{(j+0.5)L}{N}$ ($j = 0, \dots, N-1$) interpoliert. Dies soll wieder eine Approximation der Fourierkoeffizienten a_k liefern.

Nach Einsetzen der Interpolationsbedingungen erhalten wir in Vektordarstellung:

$$\underbrace{\begin{pmatrix} y_1 \\ \vdots \\ y_j \\ \vdots \\ y_{N-1} \end{pmatrix}}_y = \frac{\hat{a}_0}{2} \underbrace{\begin{pmatrix} 1 \\ \vdots \\ 1 \\ \vdots \\ 1 \end{pmatrix}}_{w_0} + \dots + \hat{a}_k \underbrace{\begin{pmatrix} \cos \frac{0.5k\pi}{N} \\ \vdots \\ \cos \frac{(j+0.5)k\pi}{N} \\ \vdots \\ \cos \frac{(N-0.5)k\pi}{N} \end{pmatrix}}_{w_k} \dots + \hat{a}_{N-1} \underbrace{\begin{pmatrix} \cos \frac{0.5(N-1)\pi}{N} \\ \vdots \\ \cos \frac{(j+0.5)(N-1)\pi}{N} \\ \vdots \\ \cos \frac{(N-0.5)(N-1)\pi}{N} \end{pmatrix}}_{w_{N-1}} \quad (9)$$

Und analog zur diskreten Fouriertransformation gilt hier auch das folgende Lemma:

Lemma 2.1. Die Vektoren $\{w_k, k = 0, \dots, N-1\}$ sind orthogonal bezüglich des Standardkalarproduktes auf dem \mathbb{R}^N mit

$$\langle w_k, w_l \rangle = \begin{cases} N, & k = l = 0 \\ \frac{N}{2}, & k = l \neq 0 \\ 0, & k \neq l \end{cases}$$

Beweis. Der Fall $k = l = 0$ ist trivial, im Fall $k = l \neq 0$ benötigen wir folgende trigonometrische Identität, welche sich aus zweimaligem Anwenden und Summieren der Additionstheoreme ergibt

$$\cos(x) \cos(y) = \frac{\cos(x+y) + \cos(x-y)}{2} \quad (10)$$

Damit folgt

$$\langle w_k, w_l \rangle = \sum_{j=0}^{N-1} \cos \frac{(j+0.5)k\pi}{N} \cos \frac{(j+0.5)l\pi}{N} \quad (11)$$

$$= \frac{1}{2} \sum_{j=0}^{N-1} \cos \frac{(j+0.5)(k+l)\pi}{N} + \underbrace{\frac{1}{2} \sum_{j=0}^{N-1} \cos \frac{(j+0.5)(k-l)\pi}{N}}_{\frac{N}{2}} \quad (12)$$

Bei der zweiten Summe ist jeder Summand 1 wegen $k = l$ und für die erste Summe verwenden wir eine ähnliche Identität wie zuvor:

$$\cos(j+0.5)x \sin \frac{x}{2} = \frac{\sin(j+1)x - \sin jx}{2}$$

Diese kann man für $x \notin 2\pi\mathbb{Z}$ umformen zu

$$\cos(j+0.5)x = \frac{\sin(j+1)x - \sin jx}{2 \sin \frac{x}{2}} \quad (13)$$

Setze $x = \frac{(k+l)\pi}{N}$ und wir erhalten aus (12) (beachte, dass $0 < \frac{(k+l)\pi}{N} < 2\pi$)

$$\begin{aligned} \langle w_k, w_l \rangle &= \frac{N}{2} + \sum_{j=0}^{N-1} \frac{\sin(j+1)x - \sin jx}{4 \sin \frac{x}{2}} \\ &= \frac{N}{2} + \frac{\sin(k+l)\pi - \sin 0}{4 \sin \frac{x}{2}} = \frac{N}{2} \end{aligned}$$

da in der Teleskopsumme nur das erste und letzte Glied, welche beide Null sind, übrig bleiben. Im Fall $k \neq l$ gehen wir analog vor, nur dass diesmal die zweite Summe in (12) nicht mehr $\frac{N}{2}$ ergibt. Es kann in diesem Fall aber auch die Identität (13) verwendet werden, womit jene Summe ebenfalls Null beträgt. \square

Somit erhält man wieder leicht die Formel für \hat{a}_k :

$$\hat{a}_k = \frac{2}{N} \langle y, w_k \rangle = \frac{2}{N} \sum_{j=0}^{N-1} y_j \cos \frac{(j+0.5)k\pi}{N} \quad k = 0, \dots, N-1$$

wobei die Formel für \hat{a}_0 trotz anderer Norm von w_0 auch gilt, da wir den Koeffizienten als $\frac{\hat{a}_0}{2}$ definiert hatten. Sei $\hat{a} := (\hat{a}_0, \dots, \hat{a}_k, \dots, \hat{a}_{N-1})$, dann nennt man die Abbildung

$$\mathcal{D}_N : \mathbb{R}^N \mapsto \mathbb{R}^N \\ y \mapsto \hat{a}$$

diskrete Kosinustransformation II (DCT II). Es gilt ein zu Lemma 1.5 analoges Resultat über die Güte der Approximation der exakten Fourierkoeffizienten und analoge Eigenschaften wie in Abschnitt 1.3 (siehe [5]). Zusammenfassend wurde Folgendes gezeigt:

1. Das gerade trigonometrische Polynom $p(t) = \frac{\hat{a}_0}{2} + \sum_{k=1}^{N-1} \hat{a}_k \cos \frac{k\pi t}{L}$ erfüllt $p\left(\frac{(j+0.5)L}{N}\right) = y_j$ für reelle y_j ($j = 0, \dots, N-1$) genau dann, wenn $\hat{a}_k = \frac{2}{N} \sum_{j=0}^{N-1} y_j \cos \frac{(j+0.5)k\pi}{N}$ (d.h. \hat{a}_k sind Bilder der DCT II)
2. Die \hat{a}_k beschreiben somit die Frequenzen des diskreten Eingangssignals $\{y_j, j = 0, \dots, N-1\}$:

$$p(t) = \underbrace{\frac{\hat{a}_0}{2}}_{\substack{\text{konstanter Term} \\ \text{Frequenz: } 0}} + \underbrace{\hat{a}_1 \cos \frac{\pi t}{L}}_{\substack{\text{Grundschwingung} \\ \text{Frequenz: } \frac{1}{L}}} + \dots + \underbrace{\hat{a}_k \cos \frac{k\pi t}{L} + \dots}_{\substack{\text{Oberschwingungen} \\ \text{Frequenz: } \frac{k}{L}}}$$

Man spricht bei den $\{\hat{a}_k, k = 0, \dots, N-1\}$ vom Frequenzspektrum des Eingangssignals.

3. Die \hat{a}_k kann man mathematisch auch als Koordinaten von y bezüglich der orthogonalen Basis $\{w_0, \dots, w_{N-1}\}$ auffassen (vgl. mit (9)). Wenn man die Vektoren in Graustufen darstellt wie in Abbildung 3, erkennt man erneut, dass die Koeffizienten den Frequenzanteil im Vektor y widerspiegeln.

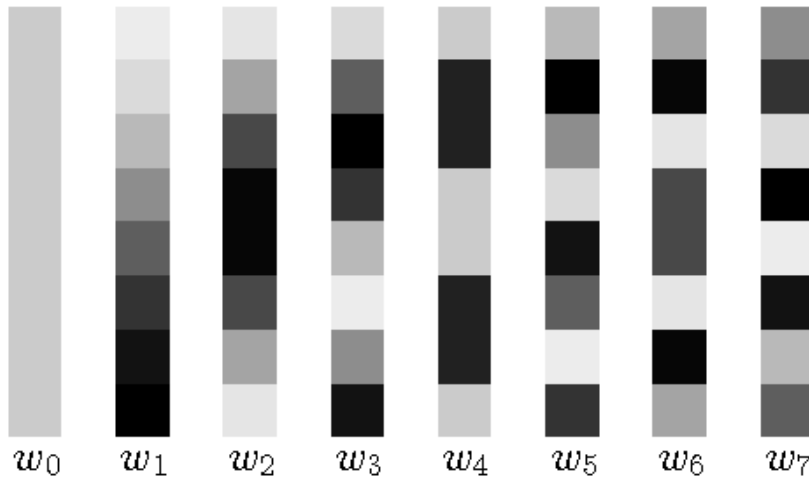


Abbildung 3: Darstellung der Basisvektoren der DCT für $N = 8$

4. In Matrixschreibweise erhalten wir:

$$\begin{aligned} \begin{pmatrix} \hat{a}_1 \\ \vdots \\ \hat{a}_k \\ \vdots \\ \hat{a}_{N-1} \end{pmatrix} &= \frac{2}{N} \underbrace{\left[\cos \frac{(j+0.5)k\pi}{N} \right]_{k,j}}_{:=D} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_k \\ \vdots \\ y_{N-1} \end{pmatrix} \\ &= \frac{2}{N} \begin{pmatrix} \overline{v_0^\top} \\ \vdots \\ \overline{v_k^\top} \\ \vdots \\ \overline{v_{N-1}^\top} \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_k \\ \vdots \\ y_{N-1} \end{pmatrix} \end{aligned}$$

wobei wie bei der DFT die Basiswechselmatrix D oft auch mit anderen Normierungen versehen wird. Die passende Normierung, um die Transformation bzw. Matrix orthogonal zu machen, ist $\sqrt{\frac{1}{N}}$ für die erste Zeile und $\sqrt{\frac{2}{N}}$ für die übrigen Zeilen (siehe Lemma 2.1).

2.2 Übersicht weiterer Kosinustransformationen

Wie der Name Kosinustransformation II und die Verschiebung der Abtaststellen um 0.5 schon nahelegen, gibt es noch weitere Kosinustransformationen. Für die Herleitung wird nun neben der trigonometrischen Interpolation eine weitere Methode nach [7] vorgestellt, bei der alle Versionen der diskreten Kosinustransformation auf natürliche Weise auftreten.

Betrachten wir diese Methode vorerst anhand der diskreten Fouriertransformation: Wir gehen wieder von der Fourierreihe aus und beobachten, dass die Basisfunktionen bei der Fourierreihenentwicklung $e_k(t) = e^{\frac{2\pi i k t}{L}}$ alle Eigenfunktion des Operators $\mathfrak{A} : u \mapsto -u''$ sind. Wenn wir diesen Operator diskretisieren, werden wir sehen, dass die zugehörigen Eigenvektoren genau die Basisvektoren der DFT sind. Für die Diskretisierung der zweiten Ableitung benötigen wir folgendes Lemma:

Lemma 2.2 (Zweite Differenzen). *Sei $u \in C^4(\mathbb{R})$, d.h. viermal stetig differenzierbar. Dann gilt:*

$$\left| \frac{u(t+h) - 2u(t) + u(t-h)}{h^2} - u''(t) \right| \leq \frac{h^2}{12} \|f^{(4)}\|_\infty \quad (14)$$

Dies bedeutet also, dass wir die zweite Ableitung an einer Stelle t durch die Funktionswerte an den Stellen $t-h, t$ und $t+h$ bis auf $O(h^2)$ abschätzen können.

Beweis. Zweimalige Auswertung der Taylorreihenentwicklung mit Restgliedabschätzung ergibt:

$$\begin{aligned} u(t+h) &= u(t) + u'(t)h + \frac{1}{2}u''(t)h^2 + \frac{1}{6}u^{(3)}(t)h^3 + \frac{1}{24}u^{(4)}(\varepsilon_1)h^4 \\ u(t-h) &= u(t) - u'(t)h + \frac{1}{2}u''(t)h^2 - \frac{1}{6}u^{(3)}(t)h^3 + \frac{1}{24}u^{(4)}(\varepsilon_2)h^4 \end{aligned}$$

mit $\varepsilon_1, \varepsilon_2 \in [t - h, t + h]$. Summieren und Abschätzen der vierten Ableitung ergibt

$$\begin{aligned} |u(t+h) - 2u(t) + u(t-h) - u''(t)h^2| &= \left| \frac{h^4}{24} [u^{(4)}(\varepsilon_1) + u^{(4)}(\varepsilon_2)] \right| \\ &\leq \frac{h^4}{12} \|u^{(4)}\|_\infty \end{aligned}$$

□

Nehmen wir nun wieder an, dass die L -periodische Funktion $u \in C^4(\mathbb{R})$ an äquidistanten Stellen t_j ($j = 1, \dots, N-1$) im Abstand h diskretisiert bzw. abgetastet wurde, wobei die Stützwerte den Vektor $y = (y_0, \dots, y_k, \dots, y_{N-1}) \in \mathbb{R}^N$ bilden. Die Diskretisierung des Operators \mathfrak{A} , d.h. $\mathfrak{A}u(t_j) \approx (A \cdot y)_j$, ergibt folgende zweite Differenzen Matrix

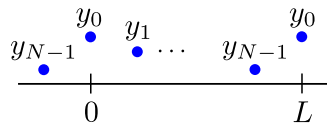
$$A = \begin{pmatrix} a_{1,1} & a_{1,2} & 0 & & & a_{1,N} \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ a_{N,1} & & & 0 & a_{N,N-1} & a_{N,N} \end{pmatrix}$$

Die Zeilen 2 bis $N-1$ ergeben sich mithilfe des Lemmas aus folgender Überlegung

$$\begin{aligned} \mathfrak{A}u(t_j) &= -u''(t_j) \approx \frac{1}{h^2} (-u(t_j-h) + 2u(t_j) - u(t_j+h)) \\ &= \frac{1}{h^2} (-y_{j-1} + 2y_j - y_{j+1}) = \frac{1}{h^2} (A \cdot y)_j \end{aligned}$$

wobei wir den Faktor $\frac{1}{h^2}$ vernachlässigen, da wir nur an den Eigenvektoren interessiert sind.

Bei den Zeilen 1 und N würden für die obige Überlegung die Werte y_{-1} und y_N benötigt werden. Die Randbedingung ergibt sich aus der Periodizität, welche unsere Funktion u erfüllen soll.



Somit muss $y_{-1} = y_{N-1}$ und $y_N = y_0$ gelten und wir erhalten folgende Matrix:

$$A = \begin{pmatrix} 2 & -1 & 0 & & & -1 \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ -1 & & & 0 & -1 & 2 \end{pmatrix}$$

Nun können wir zeigen, dass die Basisvektoren v_k der DFT tatsächlich Eigenvektoren der Matrix A sind:

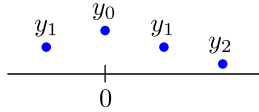
$$\begin{aligned}(A \cdot v_k)_j &= -e^{\frac{2\pi i(j-1)k}{N}} + 2e^{\frac{2\pi ijk}{N}} - e^{\frac{2\pi i(j+1)k}{N}} \\ &= \left(2 - e^{-\frac{2\pi ik}{N}} - e^{\frac{2\pi ik}{N}}\right) e^{\frac{2\pi ijk}{N}} = \left(2 - 2\cos\frac{2\pi k}{N}\right)(v_k)_j\end{aligned}$$

Damit erhalten wir automatisch auch einen neuen Orthogonalitätsbeweis (zu Lemma 1.3), da die Matrix A symmetrisch ist und somit die N unabhängigen Eigenvektoren orthogonal sein müssen.

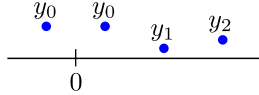
Analog gehen wir für die diskrete Kosinustransformation vor. Der Operator \mathfrak{A} bleibt gleich, nur verlangen wir diesmal als Randbedingung die Neumann-Bedingung $u'(0) = 0$, um nur Kosinusfunktionen als Eigenfunktionen zu erhalten². Am anderen Rand $t = L$ können wir wieder die Neumann-Bedingung $u'(L) = 0$ oder die Dirichlet-Bedingung $u(L) = 0$ fordern. Steigung Null führt zu den Eigenfunktionen $u_k(t) = \cos kt$, während der Funktionswert Null die Eigenfunktionen $u_k(t) = \cos(k + \frac{1}{2})t$ ergibt.

Bei der Diskretisierung bleiben also die Zeilen 2 bis $N - 1$ der Matrix A gleich und bei der ersten und letzten Zeile ergibt sich y_{-1} bzw. y_N aus der jeweiligen Randbedingung. In der ersten Zeile haben wir zwei Möglichkeiten für die Diskretisierung der Neumann-Bedingung $u'(0) = 0$:

1. 0 ist an einem Unterteilungspunkt $\Rightarrow y_{-1} = y_1$

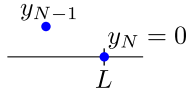


2. 0 ist Mittelpunkt zwischen zwei Unterteilungspunkten $\Rightarrow y_{-1} = y_0$

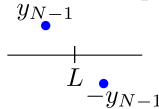


Für die letzte Zeile kann man analog die Neumann-Bedingung $u'(L) = 0$ diskretisieren oder bei der Dirichlet-Bedingung dieselbe Überlegung anstellen:

1. L ist an einem Unterteilungspunkt $\Rightarrow y_N = 0$



2. L ist Mittelpunkt zwischen zwei Unterteilungspunkten $\Rightarrow y_N = -y_{N-1}$



Insgesamt ergeben sich somit 8 Möglichkeiten für die erste und letzte Zeile der Matrix A :

²Wenn wir $u(0) = 0$ verlangen, erhalten wir die diskreten Sinustransformationen, welche aber in den Anwendungen weniger Verwendung finden.

	$y_{-1} = y_1$	$y_{-1} = y_0$
$y_N = y_{N-2}$	DCT I $\begin{pmatrix} 2 & -2 & 0 & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & 0 & -2 & 2 \end{pmatrix}$	DCT VI $\begin{pmatrix} 1 & -1 & 0 & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & 0 & -2 & 2 \end{pmatrix}$
$y_N = y_{N-1}$	DCT V $\begin{pmatrix} 2 & -2 & 0 & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & 0 & -1 & 1 \end{pmatrix}$	DCT II $\begin{pmatrix} 1 & -1 & 0 & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & 0 & -1 & 1 \end{pmatrix}$
$y_N = 0$	DCT III $\begin{pmatrix} 2 & -2 & 0 & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & 0 & -1 & 2 \end{pmatrix}$	DCT VIII $\begin{pmatrix} 1 & -1 & 0 & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & 0 & -1 & 2 \end{pmatrix}$
$y_N = -y_{N-1}$	DCT VII $\begin{pmatrix} 2 & -2 & 0 & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & 0 & -1 & 3 \end{pmatrix}$	DCT IV $\begin{pmatrix} 1 & -1 & 0 & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & 0 & -1 & 3 \end{pmatrix}$

Die Eigenvektoren dieser Matrizen sind (bis auf Skalierung) die Basisvektoren der 8 diskreten Kosinustransformationen. Die j -te Komponente des k -ten Basisvektors w_k und die zugehörige Skalierungsmatrix S ist:

$$\text{DCT-1: } \cos jk \frac{\pi}{N-1} \quad (S = \text{diag}(\sqrt{2}, 1, \dots, 1, \sqrt{2}))$$

$$\text{DCT-2: } \cos(j + \frac{1}{2})k \frac{\pi}{N} \quad (S = \text{Id})$$

$$\text{DCT-3: } \cos j(k + \frac{1}{2}) \frac{\pi}{N} \quad (S = \text{diag}(\sqrt{2}, 1, \dots, 1))$$

$$\text{DCT-4: } \cos(j + \frac{1}{2})(k + \frac{1}{2}) \frac{\pi}{N} \quad (S = \text{Id})$$

$$\text{DCT-5: } \cos jk \frac{\pi}{N-\frac{1}{2}} \quad (S = \text{diag}(\sqrt{2}, 1, \dots, 1))$$

$$\text{DCT-6: } \cos(j + \frac{1}{2})k \frac{\pi}{N-\frac{1}{2}} \quad (S = \text{diag}(1, \dots, 1, \sqrt{2}))$$

$$\text{DCT-7: } \cos j(k + \frac{1}{2}) \frac{\pi}{N-\frac{1}{2}} \quad (S = \text{diag}(\sqrt{2}, 1, \dots, 1))$$

$$\text{DCT-8: } \cos(j + \frac{1}{2})(k + \frac{1}{2}) \frac{\pi}{N-\frac{1}{2}} \quad (S = \text{Id})$$

wobei S so gewählt ist, dass die Matrix $S^{-1}AS$ symmetrisch wird. Die skalierten Vektoren $S^{-1}w_k$ sind dann Eigenvektoren der symmetrischen Matrix $S^{-1}AS$

$$A \cdot w_k = \lambda_k w_k \Rightarrow (S^{-1}AS) \cdot (S^{-1}w_k) = \lambda_k (S^{-1}w_k)$$

Damit sind die skalierten Vektoren orthogonal und bilden jeweils die Basis für die acht orthogonalen Kosinustransformationen. Jedoch sind die Kosinustransformationen I-IV, welche auf beiden Rändern dieselbe Diskretisierung besitzen (beide an Unterteilungspunkten oder Mittelpunkten), die wesentlichen Transformationen im Bereich der Datenkompression. Wir müssen noch überprüfen, dass die oben angegebenen Vektoren w_k tatsächlich Eigenvektoren der jeweiligen Matrix A sind. Wir werden dies exemplarisch anhand der DCT II vorführen, um einen alternativen Orthogonalitätsbeweis zu Lemma 2.1 zu erhalten. Für die Zeilen $j = 2, \dots, N - 1$ gilt mit $x = k \frac{\pi}{N}$

$$\begin{aligned}(A \cdot w_k)_j &= -\cos\left(j - \frac{1}{2}\right)x + 2\cos\left(j + \frac{1}{2}\right)x - \cos\left(j + \frac{3}{2}\right)x \\ &= (2 - 2\cos x)\cos\left(j + \frac{1}{2}\right)x = (2 - 2\cos x)(w_k)_j\end{aligned}$$

wobei wieder die Identität aus (10) verwendet wurde. Betrachten wir noch die Zeile $j = 1$:

$$\begin{aligned}(A \cdot w_k)_1 &= \cos\frac{1}{2}x - \cos\frac{3}{2}x = 2\cos\frac{1}{2}x - (\cos\frac{1}{2}x + \cos\frac{3}{2}x) \\ &= 2\cos\frac{1}{2}x - (\cos(1 - \frac{1}{2})x + \cos(1 + \frac{1}{2})x) \\ &= 2\cos\frac{1}{2}x - 2\cos x \cos\frac{1}{2}x = (2 - 2\cos x)\cos\frac{1}{2}x \\ &= (2 - 2\cos x)(w_k)_1\end{aligned}$$

Für die N -te Zeile und die anderen Transformationen kann man die Beweise analog führen. Betrachten wir nun ein Beispiel für die Anwendung der DCT in der Datenkompression.

2.3 DCT II in der Bildverarbeitung

Im Folgenden wird nach [8] eine Methode vorgestellt, die es erlaubt den Speicherbedarf von Bildern auf einen Bruchteil zu reduzieren ohne einen großen sichtbaren Qualitätsverlust einbüßen zu müssen. Die Methode stellt einen wichtigen Teil des Bildformates JPEG dar und beruht auf der diskreten Kosinustransformation II. Unsere Abtastwerte sind in diesem Fall die Intensität der Pixel eines Bildes, d.h. zugrunde liegt ein zweidimensionales Signal. Als Beispiel gehen wir von einem 512×512 Pixel Graustufenbild (Abbildung 4) aus, wobei die Intensität (der Grauwert) jedes Pixel in $\{0, \dots, 255\}$ liegt. In Binärdarstellung werden dafür 8 Stellen bzw. Bits verwendet, d.h. man benötigt ein Byte pro Pixel Speicherplatz.



Abbildung 4: Beispielbild



Abbildung 5: Übersicht Bildkomprimierung

2.3.1 Kodierung

Die Kodierung transformiert unser Bild in ein speichersparendes Format. Zuerst wird das Bild in (sich nicht überlappende, aneinander liegende) 8×8 -Blöcke Y aufgeteilt. Zwei Bemerkungen, auf die im Folgenden nicht genauer eingegangen wird: Falls die Pixelanzahl nicht durch 8 teilbar ist, können beispielsweise Nullen am Rand eingefügt werden. Bei Farbbildern können die jeweiligen Schritte mit den einzelnen Farbkanälen durchgeführt und die Kanäle am Schluss wieder zusammengefügt werden. Wir werden die folgenden Schritte anhand eines 8×8 -Block im Bereich der Augen (Abbildung 6) genauer betrachten.

$$Y = \begin{bmatrix} 100 & 103 & 100 & 96 & 74 & 75 & 73 & 67 \\ 69 & 70 & 87 & 84 & 64 & 64 & 67 & 62 \\ 60 & 52 & 59 & 64 & 56 & 54 & 57 & 53 \\ 53 & 51 & 54 & 52 & 51 & 52 & 52 & 49 \\ 50 & 53 & 52 & 52 & 58 & 51 & 47 & 50 \\ 53 & 53 & 51 & 53 & 55 & 51 & 53 & 45 \\ 48 & 48 & 47 & 55 & 47 & 51 & 48 & 46 \\ 48 & 53 & 53 & 47 & 43 & 54 & 49 & 50 \end{bmatrix}$$

Abbildung 6: Beispiel 8×8 -Block

Anschließend wenden wir auf jeden dieser 8×8 -Blöcke die diskrete Kosinustransformation zeilen- und spaltenweise an. Genauer bilden wir die Matrix D der DCT

für $N = 8$ (mit der Normierung, für die D orthogonal wird)

$$D = \left[\tau(k) \cos \frac{(j + 0.5)k\pi}{8} \right]_{k,j=0,\dots,7}$$

wobei die Normierung $\tau(k) = \begin{cases} \sqrt{\frac{1}{8}} & k = 0 \\ \sqrt{\frac{2}{8}} & \text{sonst} \end{cases}$

Den transformierten Block \hat{Y} erhalten wir nun durch Multiplikation des ursprünglichen Blocks Y von links mit D (spaltenweises Anwenden der DCT) und von rechts mit D^\top (zeilenweises Anwenden der DCT)

$$\hat{Y} = D \cdot Y \cdot D^\top$$

Man kann dies auch äquivalent als Transformation von $\mathbb{R}^{8 \times 8} \rightarrow \mathbb{R}^{8 \times 8}$ (zweidimensionale DCT) sehen, wobei die 8×8 Basismatrizen W_{kl} (vgl. mit w_k in (9)) folgende Gestalt haben:

$$W_{kl} = \tau(k)\tau(l) w_k \cdot w_l^\top$$

Das heißt unser 8×8 -Block Y wird in diesen 64 Basismatrizen W_{kl} dargestellt und die Koeffizienten fassen wir in der Matrix \hat{Y} zusammen.

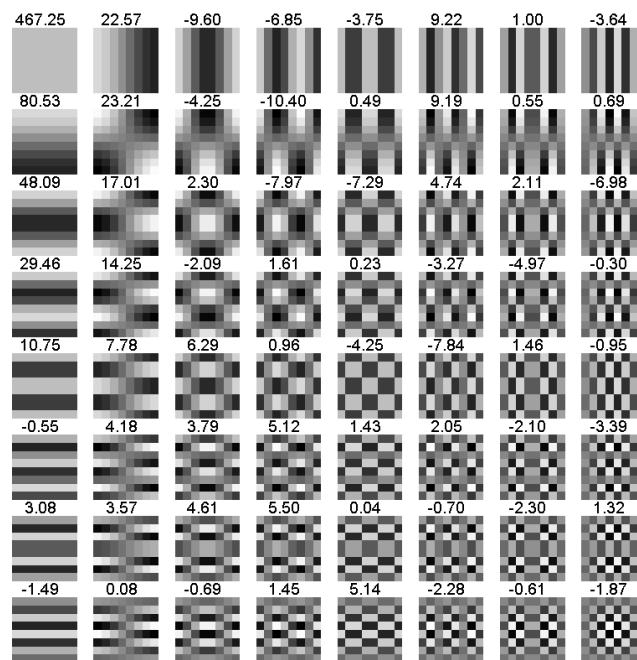


Abbildung 7: Basismatrizen und Koeffizienten des Beispielblocks

In Abbildung 7 sind die 64 Basismatrizen nicht durch Zahlen, sondern durch Graustufen dargestellt, um die Interpretation der Koeffizienten als Frequenzanteile zu motivieren. Somit spiegeln die Basismatrizen nach rechts gehend steigende horizontale und nach unten gehen steigende vertikale Frequenzen im Ausgangsbild wider.

Die Zahlen darüber sind die Koeffizienten \hat{Y} des Beispielblocks. Man erkennt exemplarisch an diesen Koeffizienten, dass die Hauptinformation des Bildausschnittes in den niedrigen Frequenzen oben links gespeichert ist. Die hohen Frequenzen unten rechts treten nur mit vergleichsweise sehr kleinen Koeffizienten auf und würden harten Kanten im Bildausschnitt entsprechen.

Im folgenden Schritt der *Quantisierung* versucht man das Bild zu komprimieren, indem man die DCT Koeffizienten nur noch mit der nötigen Präzision speichert. Das Ziel ist es lediglich die für die jeweilige Anwendung notwendige Bildinformation zu speichern. Es ist eine hochgradig nicht injektive Operation und die Hauptquelle des Datenverlusts. Ohne diesen Schritt könnte man mit der inversen diskreten Kosinustransformation II das originale Bild (bis auf Rundungsfehler) wiederherstellen.

Als Hilfsmittel dient dabei eine 8×8 -Quantisierungsmatrix Q , deren Werte entsprechend der Art des Bildes und den Ansprüchen an die Komprimierung aus $\{0, \dots, 255\}$ gewählt werden können. Wir werden als Beispiel die in Abbildung 8 dargestellte Quantisierungsmatrix benutzen, welche von der JPEG-Norm empirisch ermittelt und empfohlen wurde.

$$Q = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

Abbildung 8: Quantisierungsmatrix

Man kann jedoch auch die Matrix vervielfachen, um größere Kompression (aber auch höhere Qualitätsverluste) zu erreichen. Die Koeffizienten des transformierten Blocks (die Frequenzanteile im Bild) \hat{Y} werden nun punktweise durch die Einträge in der Quantisierungsmatrix dividiert und auf die nächste ganze Zahl gerundet:

$$\hat{Y}_{ij}^Q = \text{round} \left(\frac{\hat{Y}_{ij}}{Q_{ij}} \right) \quad \text{für } i, j = 0, \dots, 7$$

Durch das Dividieren und Runden verkleinert sich der Wertebereich umso mehr, je größer der Eintrag in der Quantisierungsmatrix ist, wodurch eine größere Spanne an Koeffizienten auf dieselben Werte geändert wird. Da die Quantisierungsmatrix unten rechts die größten Werte enthält, wird somit den hohen Frequenzen weniger Bedeutung zugeordnet. In den meisten Fällen sind die hohen Frequenzen nur mit kleinen Koeffizienten vertreten und werden auf Null gerundet. Als Ergebnis der Kodierung sieht man in Abbildung 9 den quantisierten Beispielblock. Alle quantisierten Blöcke wieder aneinander gesetzt, kann man in Abbildung 10 die Einträge ungleich Null (nur noch ca. 11%) in Blau erkennen. Man sieht hier wieder, dass die meisten Blöcke nur oben links (niedrige Frequenzen) noch Einträge ungleich Null besitzen, außer an den harten Kanten des Bildes.

$$\hat{Y}^Q = \begin{bmatrix} 29 & 2 & -1 & 0 & 0 & 0 & 0 & 0 \\ 7 & 2 & 0 & -1 & 0 & 0 & 0 & 0 \\ 3 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Abbildung 9: quantisierter Beispielblock

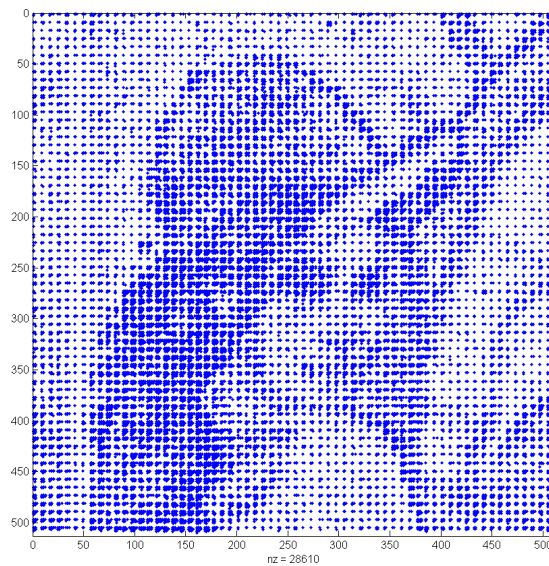


Abbildung 10: Einträge ungleich Null

Durch die Vielzahl an Nullen bzw. gleichen Werten kann die Bildinformation unter Benützung eines geeigneten Algorithmus (z.B. Zick-Zack Umsortierung und Huffman-Coding, siehe [8]) nun effizient gespeichert werden. Insgesamt kann man ungefähr davon ausgehen, dass der Anteil an Nullen auch der Einsparung an Speicherplatz entspricht.

2.3.2 Dekodierung

Im Folgenden soll das Bild aus dem speichersparende Format (bis auf Qualitätsverluste) rekonstruiert werden. Zuerst wird, bis auf die durch das Runden verloren gegangene Information, die Quantisierung in jedem 8×8 -Block wieder rückgängig gemacht, d.h. die Werte ungefähr auf den ursprünglichen Wertebereich gebracht:

$$\hat{Z}_{ij} = \hat{Y}_{ij}^Q Q_{ij} \quad \text{für } i, j = 0, \dots, 7$$

Anschließend wird die inverse diskrete Kosinustransformation angewendet (beachte, dass aufgrund der Orthogonalität $D^{-1} = D^\top$ gilt)

$$Z = D^\top \cdot \hat{Z} \cdot D$$

und für die Anzeige als Bild werden die Werte der Blöcke abschließend noch auf den Wertebereich $\{0, \dots, 255\}$ gerundet.

Anhand unseres Beispielblocks sehen wir in Abbildung 11, dass wir nur noch 11 von 64 Einträgen speichern müssen, um am Ende fast dieselben Werte rekonstruieren zu können.

$$\begin{array}{c}
 \begin{bmatrix} 100 & 103 & 100 & 96 & 74 & 75 & 73 & 67 \\ 69 & 70 & 87 & 84 & 64 & 64 & 67 & 62 \\ 60 & 52 & 59 & 64 & 56 & 54 & 57 & 53 \\ 53 & 51 & 54 & 52 & 51 & 52 & 52 & 49 \\ 50 & 53 & 52 & 52 & 58 & 51 & 47 & 50 \\ 53 & 53 & 51 & 53 & 55 & 51 & 53 & 45 \\ 48 & 48 & 47 & 55 & 47 & 51 & 48 & 46 \\ 48 & 53 & 53 & 47 & 43 & 54 & 49 & 50 \end{bmatrix} \\
 \xrightarrow{\text{Kodierung}} \\
 \begin{bmatrix} 29 & 2 & -1 & 0 & 0 & 0 & 0 & 0 \\ 7 & 2 & 0 & -1 & 0 & 0 & 0 & 0 \\ 3 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\
 \xrightarrow{\text{Dekodierung}} \\
 \begin{bmatrix} 96 & 100 & 100 & 93 & 82 & 73 & 71 & 72 \\ 74 & 78 & 80 & 76 & 68 & 62 & 61 & 63 \\ 54 & 58 & 61 & 60 & 56 & 53 & 54 & 55 \\ 51 & 53 & 55 & 56 & 55 & 53 & 53 & 53 \\ 54 & 54 & 54 & 55 & 55 & 54 & 51 & 49 \\ 53 & 51 & 50 & 50 & 52 & 51 & 46 & 42 \\ 50 & 47 & 45 & 47 & 51 & 51 & 46 & 41 \\ 48 & 45 & 43 & 47 & 53 & 55 & 51 & 45 \end{bmatrix}
 \end{array}$$

Abbildung 11: Übersicht anhand des Beispielblocks

In Abbildung 12 kann man (in Vergrößerung) das Originalbild mit dem wiederhergestellten Bildern unter Verwendung verschiedener Vielfacher der Quantisierungsmatrix vergleichen (oben rechts die von uns verwendete).

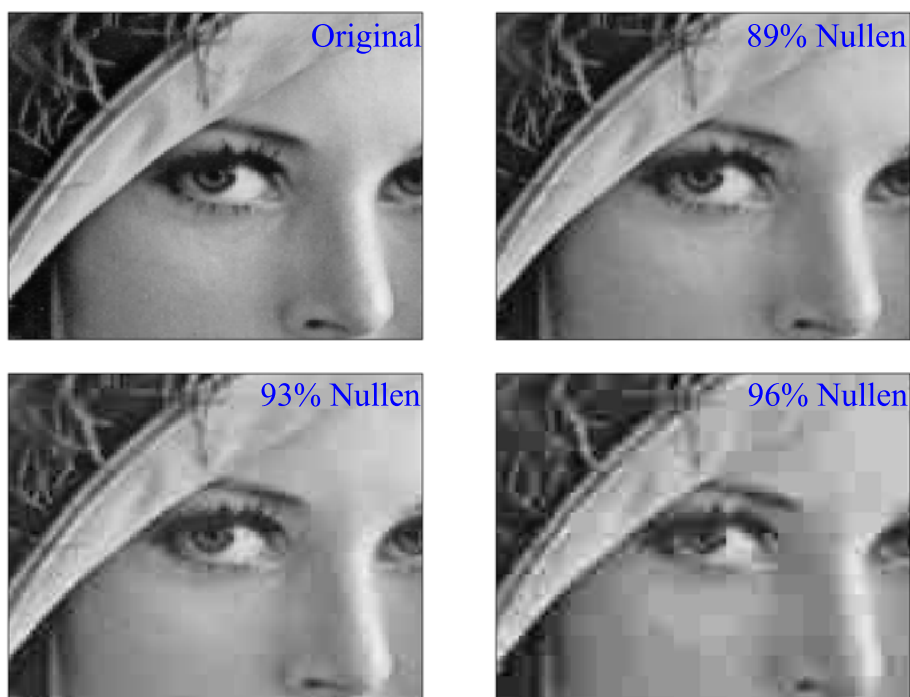


Abbildung 12: Vergleich Originalbild und decodiertes Bild

Bei unserer Quantisierungsmatrix (89% Nullen) erkennt man fast noch keine Artefakte, bei 93% Nullen sind die Kanten schon leicht verschwommen (man sieht stufige Übergänge) und bei 96% erkennt man deutliche Blockbildungen.

3 Shannon Sampling Theorem

In Kapitel 1 wurde bereits auf das Abtasten (Sampling) eines Signals eingegangen. Im Folgenden soll gezeigt werden, dass bei genügend großer Abtastfrequenz ein Signal $u(t) \in L_2(\mathbb{R})/L_1(\mathbb{R})$, das nur beschränkte Frequenzen enthält, wieder aus den Samplingwerten rekonstruiert werden kann. Dafür dient die Fouriertransformation als Hilfsmittel.

3.1 Grundlagen der Fouriertransformation

Die folgenden Sätze werden weitgehend ohne Beweis vorgestellt. Sie sind jedoch Standardresultate aus der Fourier Analysis und können beispielsweise in [2, 3, 6] nachgelesen werden.

Definition 3.1 (L_1 -Fouriertransformation). Sei $u \in L_1(\mathbb{R})$, dann definieren wir die Fouriertransformierte

$$\widehat{u}(\xi) := \int_{\mathbb{R}} u(t)e^{-2\pi i t \xi} dt \quad \forall \xi \in \mathbb{R}$$

Außerdem definieren wir die inverse Fouriertransformierte

$$u^\vee(\xi) := \widehat{u}(-\xi) = \int_{\mathbb{R}} u(t)e^{2\pi i t \xi} dt \quad \forall \xi \in \mathbb{R}$$

Die Definition ist für alle $\xi \in \mathbb{R}$ wohldefiniert, da $u \in L_1(\mathbb{R})$ und somit

$$\int_{\mathbb{R}} |u(t)e^{-2\pi i t \xi}| dt = \int_{\mathbb{R}} |u(t)| dt < \infty$$

Satz 3.2 (Eigenschaften der L_1 -Fouriertransformation). Sei $u \in L_1(\mathbb{R})$.

1. (Riemann-Lebesgue) Es gilt $\widehat{u} \in L_\infty(\mathbb{R})$ gleichmäßig stetig.
2. (L_1 -Inversionsformel) Falls auch die Fouriertransformierte \widehat{u} in $L_1(\mathbb{R})$ liegt, gilt:

$$u(t) = \widehat{u}(t)^\vee = \int_{\mathbb{R}} \widehat{u}(\xi)e^{2\pi i t \xi} d\xi \quad (15)$$

für alle t mit $u(t)$ stetig.

Bemerkung 3.3.

1. Die Darstellung in (15) motiviert die folgende Interpretation: Der Funktionswert der Fouriertransformierten \widehat{u} an der Stelle ξ gibt den Anteil der Frequenz ξ in der Funktion u an.
2. Die Fouriertransformierte von $u \in L_1(\mathbb{R})$ liegt im Allgemeinen nicht wieder in $L_1(\mathbb{R})$, wie das folgende Beispiel zeigt

Beispiel 3.4. Sei $a > 0$. Betrachten wir die Rechtecksfunktion $\text{rect}_a(t) = \chi_{[-\frac{a}{2}, \frac{a}{2}]}(t) \in L_1(\mathbb{R})$, dann gilt:

$$\begin{aligned} \widehat{\text{rect}_a}(\xi) &= \int_{\mathbb{R}} \chi_{[-\frac{a}{2}, \frac{a}{2}]}(t)e^{-2\pi i t \xi} dt = \int_{-\frac{a}{2}}^{\frac{a}{2}} e^{-2\pi i t \xi} dt = \left[-\frac{e^{-2\pi i t \xi}}{2\pi i \xi} \right]_{-\frac{a}{2}}^{\frac{a}{2}} \\ &= \frac{e^{\pi i a \xi} - e^{-\pi i a \xi}}{2\pi i \xi} = \frac{\sin \pi a \xi}{\pi \xi} = a \text{sinc } a\xi \end{aligned}$$

wobei die Sinus Cardinalis Funktion (sinc-Funktion) definiert ist als

$$\operatorname{sinc} \xi := \begin{cases} \frac{\sin \pi \xi}{\pi \xi}, & \xi \neq 0 \\ 1, & \xi = 0 \end{cases}$$

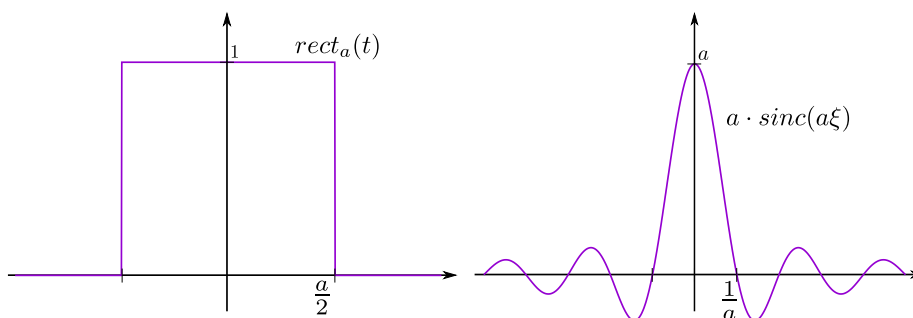


Abbildung 13: Rechtecksfunktion und Sinc-Funktion

Die sinc-Funktion ist nach L'Hospital stetig

$$\lim_{\xi \rightarrow 0} \frac{\sin \pi \xi}{\pi \xi} = \lim_{\xi \rightarrow 0} \frac{\pi \cos \pi \xi}{\pi} = 1$$

aber nicht in $L_1(\mathbb{R})$, da

$$\int_{(n-1)\pi}^{n\pi} \frac{|\sin \xi|}{|\xi|} d\xi \geq \frac{1}{n\pi} \int_{(n-1)\pi}^{n\pi} |\sin \xi| dx = \frac{2}{n\pi} \quad \text{für } n \in \mathbb{N}^+$$

Somit gilt aufgrund der harmonischen Reihe als Minorante bereits

$$\int_{\mathbb{R}^+} |\operatorname{sinc} \xi| d\xi = \infty$$

Die Definition 3.1 ist für Funktionen $u \in L_2(\mathbb{R})$ nicht mehr wohldefiniert, deshalb werden wir die Fouriertransformation mithilfe des folgenden Satzes auf L_2 -Funktionen ausdehnen.

Satz 3.5 (Plancherel). Für $u \in L_1(\mathbb{R}) \cap L_2(\mathbb{R})$, gilt:

1. $\hat{u} \in L_2(\mathbb{R})$
2. $\|\hat{u}\|_2 = \|u\|_2$

Damit können wir die Fouriertransformation für L_2 -Funktionen folgendermaßen fortsetzen: Betrachte für eine Funktion $u \in L_2(\mathbb{R})$ die Funktionen $u_n = \chi_{[-n,n]} u \in L_1(\mathbb{R}) \cap L_2(\mathbb{R})$. Nach dem Satz von der monotonen Konvergenz gilt, dass u_n eine Cauchy-Folge in $L_2(\mathbb{R})$ bildet. Daraus folgt mit Plancherel, $\|\hat{u}_n - \hat{u}_m\| = \|u_n - u_m\|$, dass $\hat{u}_n(\xi) = \int_{-n}^n u(t) e^{-2\pi i t \xi} dt$ auch eine Cauchy-Folge in $L_2(\mathbb{R})$ bildet und aufgrund der Vollständigkeit konvergiert. Dies rechtfertigt folgende Definition:

Definition 3.6 (L_2 -Fouriertransformation). Sei $u(t) \in L_2(\mathbb{R})$, dann definieren wir die Fouriertransformierte

$$\hat{u}(\xi) := \lim_{n \rightarrow \infty} \int_{-n}^n u(t) e^{-2\pi i t \xi} dt \in L_2(\mathbb{R})$$

(wobei der Limes in L_2 gemeint ist und aufgrund vorhergehender Überlegung wohldefiniert ist.)

Satz 3.7 (Eigenschaften der L_2 -Fouriertransformation). Sei $u \in L_2(\mathbb{R})$.

1. (Eindeutigkeit) Für $u \in L_1(\mathbb{R}) \cap L_2(\mathbb{R})$ stimmen die Fouriertransformierten überein. (Dies rechtfertigt im Nachhinein die Verwendung der gleichem Notation)
2. (L_2 -Inversionsformel) Es gilt

$$u \stackrel{L_2}{=} (\hat{u})^\vee$$

wobei nach Definition 3.6 die inverse L_2 -Fouriertransformation $u^\vee(t) := \hat{u}(-t)$ für $u \in L_2(\mathbb{R})$ Sinn macht.

3. (Paley-Wiener) Wir werden nur von folgender Teilaussage des Theorems von Paley-Wiener Gebrauch machen:
Falls \hat{u} kompakten Träger besitzt, gibt es einen stetigen Repräsentanten in der Äquivalenzklasse von u .

Mit dieser Vorarbeit können wir das Shannon Sampling Theorem beweisen.

3.2 Theorem und Beweis

Hierbei gehen wir von einer bandbeschränkten Funktion $u(t) \in L_2(\mathbb{R})$ aus, d.h. die Fouriertransformierte besitzt kompakten Träger $\text{supp}(\hat{u}) \subseteq [-\lambda, \lambda]$. Da nach Definition 3.6 $\hat{u} \in L_2(\mathbb{R})$ gilt, ist der Träger nur bis auf eine Nullmenge bestimmt und dementsprechend ist die Inklusion gemeint. Gemäß der Interpretation in Bemerkung 3.3 (i) kann man davon sprechen, dass die Funktion u nur Frequenzen zwischen $-\lambda$ und λ enthält. Wenn diese Funktion nun mit einer Frequenz $\omega \geq 2\lambda$ abgetastet wird (anders gesagt die Zeit zwischen zwei Abtaststellen muss kleiner gleich $\frac{1}{2\lambda}$ sein), dann kann die Funktion u aus den Abtastwerten $\{u(\frac{k}{\omega})\}_{k \in \mathbb{Z}}$ rekonstruiert werden.

Satz 3.8 (Shannon-Sampling-Theorem). Sei $u \in L_2(\mathbb{R})$ bandbeschränkt mit $\text{supp}(\hat{u}) \subseteq [-\lambda, \lambda]$ und $\omega \geq 2\lambda$, dann gilt:

$$u(t) \stackrel{L_2}{=} \sum_{k \in \mathbb{Z}} u\left(\frac{k}{\omega}\right) \text{sinc}(\omega t - k)$$

Beweis. Definiere $g := \chi_{(-\frac{\omega}{2}, \frac{\omega}{2})} \hat{u} \in L_2(-\frac{\omega}{2}, \frac{\omega}{2})$, da per Definition 3.6 $\hat{u} \in L_2(\mathbb{R})$ gilt. Nach Bemerkung 1 können wir also g in eine (in L_2) konvergente Fourierreihe entwickeln

$$g(t) \stackrel{L_2}{=} \sum_{k \in \mathbb{Z}} c_k e^{\frac{2\pi i k t}{\omega}} \tag{16}$$

mit

$$c_k = \frac{1}{\omega} \int_{-\frac{\omega}{2}}^{\frac{\omega}{2}} g(t) e^{-\frac{2\pi i k t}{\omega}} dt = \frac{1}{\omega} \int_{-\frac{\omega}{2}}^{\frac{\omega}{2}} \hat{u}(t) e^{-\frac{2\pi i k t}{\omega}} dt \quad (17)$$

$$= \frac{1}{\omega} \int_{-\infty}^{\infty} \hat{u}(t) e^{-\frac{2\pi i k t}{\omega}} dt = \frac{1}{\omega} u\left(-\frac{k}{\omega}\right) \quad (18)$$

Hier wurde die Definition von g , der Träger von \hat{u} und in der letzten Gleichheit die Inversionsformeln und die Eindeutigkeit der Fouriertransformation benützt. Beachte, dass wegen des kompakten Trägers $\hat{u} \in L_2(\mathbb{R}) \cap L_1(\mathbb{R})$ und man hier immer den aufgrund von Paley-Wiener vorhandenen stetigen Repräsentanten aus der Äquivalenzklasse u wählt.

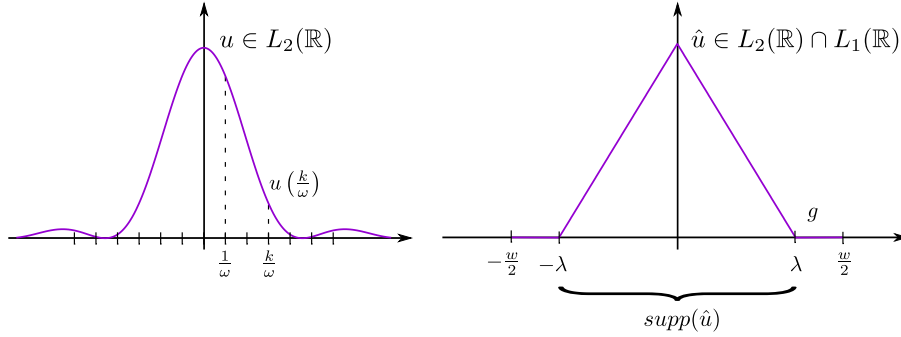


Abbildung 14: Illustration zum Beweis

Da g mit \hat{u} auf dessen Träger übereinstimmt, haben wir nun mithilfe von (16) und (18) die Fouriertransformierte von u mithilfe der Abtastwerte rekonstruiert. Nun müssen wir nur noch die inverse Fouriertransformation anwenden:

$$\begin{aligned} u(t) &\stackrel{L_2}{=} (\hat{u}(t))^\vee = \int_{\mathbb{R}} \hat{u}(\xi) e^{2\pi i \xi t} d\xi = \int_{-\frac{\omega}{2}}^{\frac{\omega}{2}} g(\xi) e^{2\pi i \xi t} d\xi \\ &= \int_{-\frac{\omega}{2}}^{\frac{\omega}{2}} \sum_{k \in \mathbb{Z}} c_k e^{\frac{2\pi i k \xi}{\omega}} e^{2\pi i \xi t} d\xi \end{aligned}$$

Nach Fubini, angewandt auf das Integral und die Summe (betrachtet als Integral über das diskrete Maß), dürfen wir jene beiden vertauschen, da $\hat{u} \in L_1(\mathbb{R})$ ist:

$$\begin{aligned} u(t) &\stackrel{L_2}{=} \sum_{k \in \mathbb{Z}} \int_{-\frac{\omega}{2}}^{\frac{\omega}{2}} c_k e^{\frac{2\pi i k \xi}{\omega}} e^{2\pi i \xi t} d\xi = \sum_{k \in \mathbb{Z}} \int_{-\frac{\omega}{2}}^{\frac{\omega}{2}} \frac{1}{\omega} u\left(-\frac{k}{\omega}\right) e^{2\pi i \xi \left(\frac{k}{\omega} + t\right)} d\xi \\ &= \sum_{k \in \mathbb{Z}} \frac{1}{\omega} u\left(-\frac{k}{\omega}\right) \left[\frac{e^{2\pi i \xi \left(\frac{k}{\omega} + t\right)}}{2\pi i \left(\frac{k}{\omega} + t\right)} \right]_{-\frac{\omega}{2}}^{\frac{\omega}{2}} = \sum_{k \in \mathbb{Z}} u\left(-\frac{k}{\omega}\right) \frac{\operatorname{Im}(e^{\pi i (\omega t + k)})}{\pi (\omega t + k)} \\ &= \sum_{k \in \mathbb{Z}} u\left(\frac{k}{\omega}\right) \frac{\sin \pi (\omega t - k)}{\pi (\omega t - k)} = \sum_{k \in \mathbb{Z}} u\left(\frac{k}{\omega}\right) \operatorname{sinc}(\omega t - k) \end{aligned}$$

□

Bemerkung 3.9.

1. Mit ein wenig mehr Aufwand lässt sich auch die gleichmäßige Konvergenz der Reihe gegen die Funktion zeigen, siehe [4].
2. Das Shannon-Sampling-Theorem gilt auch für bandbeschränkte $L_1(\mathbb{R})$ -Funktionen, da diese automatisch auch in $L_2(\mathbb{R})$ liegen müssen: Angenommen $u \in L_1(\mathbb{R})$ bandbeschränkt, dann gilt nach Riemann-Lebesgue 3.2 \hat{u} stetig und beschränkt. Da $\text{supp}(\hat{u})$ kompakt ist, können wir $\hat{u} \in L_1(\mathbb{R}) \cap L_2(\mathbb{R})$ schließen. Aus Plancherel 3.7 folgt $\|u\|_2 = \|\hat{u}\|_2 < \infty$ und damit $u \in L_2(\mathbb{R})$.

Abbildungsverzeichnis

1	komplexe Exponentialfunktion, Eigene Abbildung	4
2	Abtasten einer Funktion, Eigene Abbildung	5
3	Basisvektoren der DCT, Eigene Abbildung	12
4	Lena Testbild, USC-SIPI image database	18
5	Übersicht Bildkomprimierung, Eigene Abbildung	18
6	Beispielblock, Eigene Abbildung	18
7	Basismatrizen der DCT, Eigene Abbildung	19
8	Quantisierungsmatrix, Eigene Abbildung	20
9	quantisierter Beispielblock, Eigene Abbildung	21
10	Einträge ungleich Null, Eigene Abbildung	21
11	Übersicht anhand des Beispielblocks, Eigene Abbildung	22
12	Vergleich Originalbild und decodiertes Bild, eigene Abbildung	22
13	Rechtecks- und Sincfunktion, Eigene Abbildung	24
14	Beweisillustration, Eigene Abbildung	26

Literatur

- [1] Amann, H. (2008). Analysis II. Birkhäuser.
- [2] Bachmann, G., Narici, L., & Beckenstein, E. (2012). Fourier and wavelet analysis. Springer Science & Business Media.
- [3] Gasquet, C. (1999). Fourier Analysis and Applications. Springer.
- [4] Marks, R. (2012). Introduction to Shannon sampling and interpolation theory. Springer Science & Business Media.
- [5] Rao, K. R., & Yip, P. (2014). Discrete cosine transform: algorithms, advantages, applications. Academic press.
- [6] Stade, E. (2011). Fourier analysis (Vol. 109). John Wiley & Sons.
- [7] Strang, G. (1999). The Discrete Cosine Transform. SIAM review, 41(1), S. 135-147.
- [8] Wallace, G. K. (1992). The JPEG still picture compression standard. IEEE transactions on consumer electronics, 38(1), xviii-xxxiv.