Ludwig-Maximilians-Universität München
LFE Medieninformatik
Prof. Dr. Andreas Butz
Dipl. Inf. Otmar Hilliges

Wintersemester 05/06
Übungsblatt 5
25. November 2005

Übung 2D Grafik

Abgabetermin :
Die Lösung zu diesem Übungsblatt ist bis zum **12./14. Dezember 2005** abzugeben. Sie haben also zwei Wochen zur Bearbeitung der Übungsaufgaben Zeit.

Inhalt:
In diesem Übungsblatt behandeln wir die Fourier Transformation und ihre Anwendung in der Bildverarbeitung. Die grundlegenden Konzepte der Fourier Transformation (FT) haben sie in der Vorlesung kennen gelernt, hier sollen sie ihr wissen praktisch anwenden und vertiefen.

Aufgabe 1 (P) Diskrete Fourier Transformation

Jedes beliebige Signal, z.B. Töne im 1-dimensionalen bzw. Bilder im 2-dimensionalen, besteht aus einer Kombination von Signalen unterschiedlicher Frequenz.

Anhand eines Prismas können sie sich sehr gut veranschaulichen, wie Signale aus unterschiedlichen Signalen zusammen gesetzt sind. Wenn weißes Licht auf das Prisma trifft, wird das Signal in seine konstituierenden Signale (farbiges Licht) aufgespalten.

Analog lässt sich jede periodische Funktion durch eine gewichtete Summe von Sinusoiden (Sinus und Cosinus Wellen) ausdrücken. Diese Erkenntniss ist die Grundlage der Fouriertheorie. Die einzelnen Funktionen der Summe heißen Basisfunktionen. Die Fouriertheorie gibt uns ein Werkzeug um die Anteile jeder einzelnen Basisfunktion in der Repräsentation einer Funktion $f(x)$ zu analysieren. Für Bilder gilt, dass die Originalfunktion zweidimensional ist.

Die FT in zwei Dimensionen lässt sich folgendermaßen angeben:

$$F(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(m,n) * e^{-i*2\pi(um+vn)}\, dmdn \tag{1}$$

Analog ist die inverse FT als

$$f(m,n) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(u,v) * e^{i*2\pi(um+vn)}\, dudv \tag{2}$$

definiert.

Dabei bezeichnet $F(u,v)$ die Funktion im *Frequenzraum*, oder die Fourier-Transformierte, und $f(m,n)$ die Originalfunktion im *Ortsraum*. Beachten sie dabei, dass $F(u,v)$ eine komplexe Zahl ist. Im Frequenzraum bezeichnet u die Frequenz entlang der originalen m-Achse und v die Frequenz entlang der originalen n-Achse. Wie in Abb. 1 zu sehen, ist der Ursprung des Frequenzraumes in der Mitte des Bildes, nicht wie im Ortsraum links oben.

Wenn wir mit Bildern arbeiten, haben wir niemals eine kontinuierlich definierte Funktion sondern diskrete Samples - die Pixel des Bildes. Um mit solchen Daten trotzdem im Frequenzraum arbeiten zu können, benötigen sie einen Spezialfall der FT - die diskrete FT. Anstatt der Integration für kontinuierliche Funktionen, verwenden wir hier Summen über die diskreten Samples.

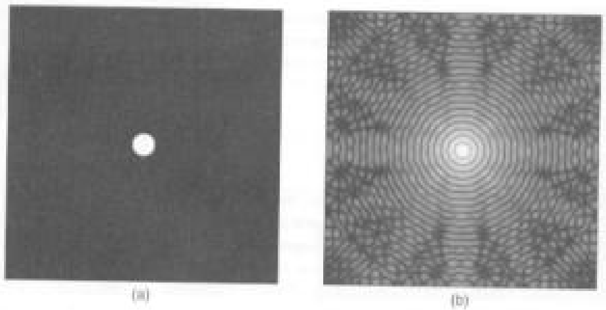

Abbildung 1: Fourier-Transformierte eines Punktes. (a) der Ortsraum. (b) der Frequenzraum

Die Formel um eine Bild der Dimension MxN in den Frequenzraum zu transformieren ist als

$$F(u,v) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f(m,n) * e^{-i*2\pi*\left(\frac{um}{M}+\frac{vn}{N}\right)} \tag{3}$$

gegeben. Die Formel um zurück in den Ortsraum zu gelangen ist

$$f(m,n) = \frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} F(u,v) * e^{i*2\pi*\left(\frac{um}{M}+\frac{vn}{N}\right)} \tag{4}$$

Der einzige Unterschied, zwischen den beiden Formeln, ist das Vorzeichen im Exponenten. Daraus lassen sich zwei Dinge ablesen. Erstens, ist die FT in beide Richtungen verlustfrei, also eine eindeutige Abbildung. Und zweitens, wird unmittelbar deutlich, dass sich der selbe Code verwenden lässt, um die FT und die inverse FT zu implementieren.

Um zu verstehen, was die Fouriertransformierte eines Bildes ausdrückt, muss man sich nochmal verdeutlichen, das $F(u,v)$ eine komplexe Zahl ist.

$$F(u,v) = R(u,v) + i * I(u,v) = |F(u,v)| * e^{i*\phi(u,v)} \tag{5}$$

Der reelle $R(u,v)$ und imaginäre $I(u,v)$ Anteil an sich haben noch keine besondere Bedeutung, allerdings ist $|F(u,v)|$ die Amplitude und $\phi(u,v)$ ihre Phase. Zerlegt man nun ein Bild im Frequenzraum in ein Array von Amplituden und ein Array von Phasen, dann erhalten wir das Amplituden-Spektrum, respektive das Phasen-Spektrum eines Bildes. Diese Spektren lassen sich wiederum als Bilder (Abb. 2) zeichnen. Das Amplituden-Spektrum gibt die relativen Helligkeiten von Objekten

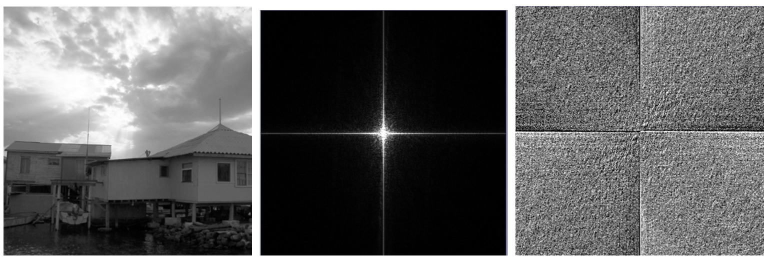

Abbildung 2: Ein Bild im Ortsraum, sein Amplituden-Spektrum, und das Phasen-Spektrum.

im Ortsraum wieder und das Phasen-Spektrum encodiert die Information über die Position dieser Objekte. Die meisten Operationen im Frequenzraum lassen deswegen das Phasen-Spektrum unverändert.

a) Machen sie sich mit den Konzepten der FT und insbesondere mit der diskreten Variante vertraut. Benutzten sie dazu z.B. die Slides der Vorlesung, das Buch *Grundlagen der Bild-*

verarbeitung[1] und, eher praktisch ausgerichtet, die Website
http://www.netnam.vn/unescocourse/computervision/91.htm.

Aufgabe 2 (P) Fast Fourier Transformation

Um ein einen Wert von $F(u,v)$ nach obiger Formel zu berechnen, muss einmal über alle Pixel summiert werden. Bei einem NxN Bild ergibt das eine Komplexität von $O(N^2)$ und folglich für alle Werte eine Gesamtkomplexität von $O(N^4)$. Das führt schon bei sehr kleinen Bildern zu nicht akzeptablen Laufzeiten.

Glücklicherweise hat die FT zwei Eigenschaften die es erlauben diese Laufzeiten deutlich zu verbessern.

Separabilität Die 2-dimensionale FT lässt sich in zwei 1-dimensionale FTs aufspalten. Zuerst wird eine 1-dimensionale FT entlang den Zeilen eines Bildes ausgeführt um ein temporäres Ergebniss zu berechnen, danach wird eine zweite FT entlang den Spalten dieses Ergebnisses durchgeführt um das entgültige Ergebniss zu erlangen.

Symmetrie Die 1-dimensionale FT der Länge N kann als Summe zweier FTs der Länge $\frac{N}{2}$ ausgedrückt werden. Ist nun N eine 2er Potenz, dann kann man diesen Schritt rekursiv anwenden, bis Transformationen der Länge 2 betrachtet werden.

Die *Fast Fourier Transformation* (FFT) ist ein *Divide-and-Conquer* Algorithmus der obige Eigenschaften ausnützt und damit die Komplexität auf $O(N^2 \log N)$ senkt. Die FFT für Bilder erfordert, dass beide Bild-dimensionen 2er Potenzen sind. Ist das nicht der Fall, muß das Bild entweder beschnitten oder mit Nullen aufgefüllt werden.

Die Fouriertheorie geht davon aus, dass die betrachteten Funktionen periodisch sind, sich also endlos wiederholen. Diese Annahme gilt allerdings für Bilder in der Regel nicht. Kachelt man ein Bild mehrfach in unterschiedliche Richtungen, sind an den Kanten starke Diskontinuitäten zu beobachten. In der FT werden die Eingabedaten aber als eine Periode einer sich unendlich wiederholenden Frequenz betrachtet. Die unpassenden Anschlußstellen an den Rändern des Bildes wirken sich wie abrupte Änderungen im Bild aus und der Frequenzraum wird verzerrt.

Um diese unerwünschte Verzerrung im Frequenzraum zu verhindern kann man die Pixelwerte des Bildes so modifizieren, dass sie sich zum Bildrand hin Null annähern. Dies geschieht indem man eine so genannte *Fenster-Funktion (engl. windowing)* zu den Daten multipliziert bevor man die FT durchführt. Es gibt mehrere Standard-Fenster, z.B. das Bartlett-Fenster is wie folgt definiert:

$$w(r) = \begin{cases} 1 - \left(\frac{r}{r_{max}}\right) & ,r \leq r_{max} \\ 0 & ,r > r_{max} \end{cases} \tag{6}$$

wobei r der Abstand vom Bildmittelpunkt ist. Das Hannning-Fenster, definiert für $r \leq r_{max}$

$$w(r) = \frac{1}{2} - \frac{1}{2} cos * \left[\pi * \left(1 - \frac{r}{r_{max}}\right)\right] \tag{7}$$

a) Machen sie sich mit dem FFT Algorithmus und seiner Implementierung vertraut. Gute Ausgangspunkte sind der Wikipedia Eintrag zum originalen FFT
http://en.wikipedia.org/wiki/Cooley-Tukey_FFT_algorithm, und die beiden folgenden Webseiten
http://www.dspguru.com/info/faqs/fftfaq2.htm
http://www.fftw.org/links.html

[1]Klaus.D.Tönnies, ISBN 3-8273-7155-4

b) Auf der Übungsseite finden sie ein JAR-file mit den Klassen *FastFourierTrans.java*, *FFTException.java* und *Complex.java*. In diesen Klassen finden sie ein Codegerüst um ihren eigenen FFT Algorithmus zu implementieren. Alle Methoden um das Bild zu kopieren, die Daten vorzubereiten und um mit komplexen Zahlen umzugehen sind bereits vorbereitet. Desweiteren finden sie Methoden um die in (6), (7) vorgestellten Fensterfunktionen anzuwenden. Zuletzt finden sie Methoden um das Amplituden- sowie Phasen-Spektrum als BufferedImage auszugeben.

c) Vervollständigen sie die Methoden *transform()* für den Divide-Schritt, und *fft()* für den *Conquer* Schritt des FFT Divide-and-Conquer Verfahrens.

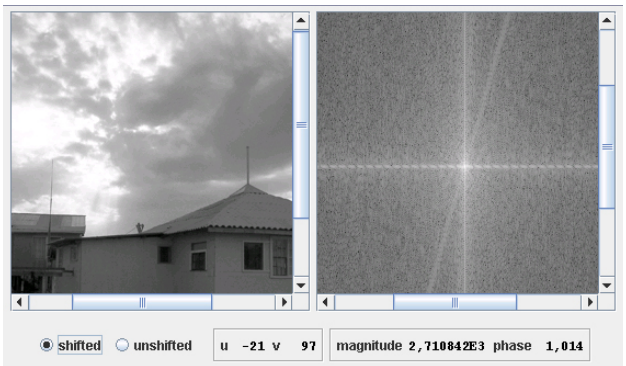

Abbildung 3: Spektrum Inspektor

d) Implementieren sie einen Spektrumsinspektor ähnlich Abb. 3 um Bilder im Ortsraum und im Frequenzraum anzeigen zu können. Achten sie auch darauf, dass das Amplituden-Spektrum und das Phasen-Spektrum getrennt angezeigt werden kann.