

Das Gibbs'sche Phänomen

von Jürgen Müller

Thema: Digitalisierung und das Gibbs'sche Phänomen

Übersicht

Klassenstufe	Sekundarstufe II
Oberthemen	ITG: – Informatik: Algorithmen, Visualisierung und grafische Datenverarbeitung
Unterthemen	ITG: – Informatik: Grafikformate, Digitalisierung, Kompressionsalgorithmen
Anforderungsniveau	mittel bis hoch
Durchführungsniveau	gering
Vorwissen	Physik: Harmonische Schwingungen
Methode	Schülerexperiment
Vorbereitung	Je Arbeitsplatz 5 Minuten für die Installation
Durchführung	15 Minuten

Software und Materialien

Für dieses Experiment wird Software zur Bildbearbeitung benötigt, mit der Bilddateien gezoomt werden können, mit der sich Text in das Bild einfügen lässt und die eine Bildkonvertierung in das JPEG-Format ermöglicht. Die meisten Grafikprogramme verfügen über diese Funktionen. Im vorliegenden Beitrag wird die französische Freeware *PhotoFiltre* in deutscher Lokalisierung genutzt, die ausreichend leistungsfähig sowie schnell zu installieren ist und auch auf Computern einer schwächeren Leistungsklasse läuft. Die Bezugsquelle ist am Ende des Beitrages angegeben; darüber hinaus steht *PhotoFiltre* über den LOG-IN-Service zur Verfügung (s. S. 76).

Weiterhin werden Digitalfotos im JPEG-Format benötigt, die möglichst harte Motivübergänge beinhalten, wie z.B. Bäume oder Gebäude vor einem hellen Himmels-hintergrund. Diese Fotos kann die Lehrkraft bereitstellen; besser ist es allerdings, wenn die Schülerinnen und Schüler jeweils zwei bis drei eigene Fotos auf Diskette oder USB-Datenträger mitbringen. Digitalkameras bieten in der Regel eine Speicherung der Aufnahmen im JPEG-Format. Geeignete Bilddateien finden sich auch im LOG-IN-Service (s. S. 76) zum Herunterladen.

Vorbereitung

Auf den Schülerarbeitsplätzen muss die Bildbearbeitungssoftware installiert sein. Die mitgebrachten JPEG-Bilddateien werden von den Schülerinnen und Schülern auf die Festplatte ihrer Rechner kopiert. Die Schülerinnen und Schüler sollten von der Lehrkraft mit den elementaren Funktionen der Bildbearbeitungssoftware vertraut gemacht werden (Bilder laden, Bilder neu erstellen, Zoomen der Bilder, Einfügen von Text und anderen grafischen Elementen, Speichern der Bilder in verschiedenen Grafikformaten). Gegebenenfalls kann eine entsprechende Unterweisung während der Durchführung des Versuchs erfolgen.

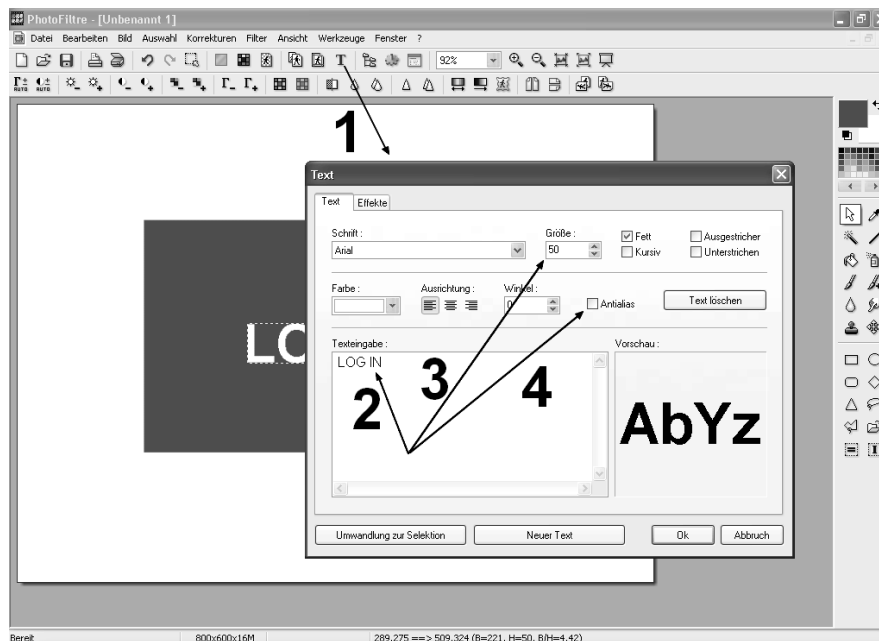
Durchführung

Folgende Arbeitsanweisungen werden gegeben:

1. Erstellt mit der Grafiksoftware ein neues Bild (Bildgröße: 800 × 600 Punkte)!
2. Zeichnet ein rotes Rechteck in das Bild (Werkzeugeleiste am rechten Rand von *PhotoFiltre*, Rechteck zeichnen, Farbe durch Anklicken wählen, mit Füllwerkzeug in Rechteck klicken, um es mit der gewählten Farbe zu füllen)!
3. Fügt mit weißer Farbe Text in der Größe von 50 Punkt ein, z.B. euren Vornamen (vgl. Anleitung in Bild 1, nächste Seite)! Beachtet dabei, dass das Häkchen in der *Checkbox* Antialiasing nicht gesetzt ist!
4. Speichert das Bild im BMP-Format (Menüpunkt *Speichern unter*; im Auswahlfeld *Dateityp* das entsprechende Format wählen)!
5. Speichert das Bild dann im JPEG-Format! Wählt für die Kompression einen Wert von etwa 30 (Bild 2, nächste Seite)!
6. Ladet das eben gespeicherte Bild im BMP-Format und zoomt es mit einer Vergrößerung von etwa 1000! Stellt die Ansicht so ein, dass ein Teil des Textes sichtbar ist!
7. Ladet als nächstes das Bild im JPEG-Format und zoomt es ebenfalls mit einer Vergrößerung von etwa 1000! Stellt auch hier die Ansicht so ein, dass ein Teil des Textes sichtbar ist!
8. Vergleicht beide Darstellungen! Stellt Gemeinsamkeiten und Unterschiede fest!
9. Stellt die Größen der beiden Bilder in kB (Kilobyte) fest!

Bilder 1 und 2: Verwendet wird für das Experiment die Bildbearbeitungssoftware PhotoFiltre (Freeware). Bild 1 zeigt die Reihenfolge beim Einfügen von Text in ein Bild. Wichtig ist, das Antialiasing zunächst nicht zu verwenden, da dadurch die Beobachtung des Gibbs'schen Phänomens erschwert wird.

Beim Konvertieren des Bildes in das JPEG-Format sollte ein Kompressionswert von ungefähr 30 gewählt werden, damit das Gibbs'sche Phänomen deutlich zu erkennen ist (Bild 2).



Damit ist der Treppenstufeneffekt also die Konsequenz der Digitalisierung; er tritt bei jeder Form der Umwandlung von analogen in digitale Daten auf und zeigt sich somit auch bei digitalen Audiodaten wie dem MP3-Format, bei digitalisierten Videos usw.

Der Treppenstufeneffekt kann durch höhere Bildauflösung und durch sogenanntes *Antialiasing* gemindert werden.

Unterschiedliche Dateigrößen bei verschiedenen Grafikformaten

Der Vergleich der Dateigrößen beider Bildformate zeigt deutliche Unterschiede: Während das Bild im BMP-Format fast 1400 kB groß ist, beansprucht das Bild im



Beobachtung – Funktionsweise – Erklärung

Gemeinsamkeit: Treppenstufeneffekt

Beide Bilder zeigen den sogenannten Treppenstufeneffekt: An den Kanten der Schrift bilden die Pixel (engl. Kunstwort für *picture element*) ein treppenartiges Erscheinungsbild (Bild 3). Der Treppenstufeneffekt wird bei einer Umwandlung des Bildes in ein Bitmap-Format durch die endliche Bildauflösung des Monitors (oder allgemeiner des entsprechenden Grafikgeräts wie Digitalkamera, Scanner, Drucker usw.) verursacht. Um ein Bild zu digitalisieren, wird das Bild in Zeilen und Spalten zerlegt (gescannt), für jeden der dadurch entstehenden Bildpunkte der Farbwert ausgelesen und mit einer bestimmten Quantisierung gespeichert. Bei der *Quantisierung* wird der ursprünglich kontinuierliche Wertebereich (z.B. eine beliebiger Farbwert) auf einer diskreten Menge (z.B. Zahlen zwischen 0 und 255) abgebildet.

Bild 3: Der durch die Digitalisierung verursachte Treppenstufeneffekt tritt bei komprimierten und unkomprimierten Grafikformaten gleichermaßen auf. Das Gibbs'sche Phänomen dagegen ist nur bei der JPEG-Grafik zu sehen (rechtes Bild).



Bild 4: Die beobachtbaren Pixelfehler haben ihre Ursache in der Fourieranalyse, die im Rahmen der Bildkompression beim JPEG-Format eingesetzt wird.

JPEG-Format nur den zehnten Teil dieses Speicherplatzes und kommt mit 14 kB aus. Beide Bilder zeigen in „Normalansicht“ (Zoomstufe 100 % bei PhotoFiltre) auf den ersten Blick keine erkennbaren Unterschiede.

Zur Speicherung des digitalisierten Bildes können gegebenenfalls Komprimierungs-Formate eingesetzt werden. Digitalisierte Bilder können recht groß werden, eine Übertragung z.B. über das Internet würde dann sehr lange dauern. Ein Rechenbeispiel: Die erzeugten Bilder haben eine Größe von 800×600 Bildpunkten. Das Grafikformat BMP (Windows Bitmap) erlaubt Quantisierungen von 1, 4, 8, 16, 24 oder 32 Bit je Pixel (bpp). Bei PhotoFiltre werden 24 bpp verwendet.

Damit berechnet sich die Größe des Bildes in kB wie folgt:

$$\text{Größe} = \frac{\text{Länge} \times \text{Breite} \times \text{Quantisierung}}{8 \times 1024} \text{ kB}$$

Für das im Versuch erzeugte Bild ergibt sich damit die Größe von

$$\text{Größe} = \frac{800 \times 600 \times 24}{8 \times 1024} \text{ kB} = 1406,25 \text{ kB}$$

Das bedeutet, dass auf einer 3,5-Zoll-Diskette gerade einmal ein Bild gespeichert werden kann! Der Vergleich mit der Datei im JPEG-Format (Größe: 14 kB) zeigt aber auch, dass offensichtlich die resultierende Dateigröße durch geeignete Algorithmen drastisch reduziert werden kann. Die Bildqualität muss darunter nicht leiden, denn eine Betrachtung beider Bilder zeigt – wie schon erwähnt – keine Unterschiede.

Fourieranalyse und das Gibbs'sche Phänomen

Der Vergleich der beiden Grafikformate ergibt noch einen weiteren Unterschied: Während bei der BMP-Datei die Farben wie im Original erhalten sind und nur der Treppeneffekt stört, zeigen sich beim JPEG-Format an den Kanten des Textes „Verunreinigungen“ (Bild 3, vorige Seite, und Bild 4). Beim Übergang vom weißen Text zum roten Hintergrund sind entlang der Kanten Pixel in anderen Farben entstanden. Im weißen wie im roten Bereich tauchen dunklere Pixel auf, im roten Bereich auch einige hellere, verglichen mit der Umgebungsfarbe der Pixel (Bild 4).

Diese Erscheinung wird nach seinem Entdecker, dem Physiker **Josiah Willard Gibbs** (1839–1903), als das Gibbs'sche Phänomen bezeichnet. Damit wird das Verhalten von Fourierreihen in der Umgebung von Sprungstellen charakterisiert. Fourierreihen werden wegen ihrer besonderen Eigenschaften gerne für die Kompression von Bild- und Sprachsignalen verwendet. Eine ausführliche Erläuterung kann an dieser Stelle nicht erfolgen, verwiesen sei auf das Buch von Meffert und Hochmuth (Meffert/Hochmuth, 2004). Folgende Plausibilitätserklärung mag ausreichen und kann auch den Schülerinnen und Schülern vermittelt werden. Der französische Mathematiker und Physiker Jean Baptiste Joseph Fourier (1768–1830) entdeckte 1807, „... dass sich jedes periodische Signal in harmonische Bestandteile zerlegen lässt, deren Komponenten sich in Amplitude, Phase und Frequenz unterscheiden, wobei die Frequenz der Komponenten immer ein Vielfaches der Grundfrequenz ist“ (Meffert/Hochmuth, 2004, S.137). Im Bild 5 werden zur Verdeutlichung ein periodisches Signal und seine harmonischen Bestandteile gezeigt. Wie Fourier nachweisen konnte, lässt sich jede periodische Schwingungsform in eine Additionsreihe zerlegen, deren einzelne Glieder aus einfachen Sinusschwingungen bestehen. Diese Tätigkeit der Zerlegung komplexer periodischer Schwingungen in ihre aus Sinusschwingungen bestehenden Komponenten nennt man *Fourieranalyse*.

Der Algorithmus zur Erzeugung von JPEG-Grafiken führt eine solche Fourieranalyse (genauer: eine Diskrete Kosinustransformation) auf den Bilddaten durch. Dazu wird zunächst das unkomprimierte Bild so aufbereitet, dass Dateneinheiten zu 8×8 Pixel erzeugt werden. Jedes Pixel darin kann den Wert einer ganzen Zahl zwischen 0 und 255 annehmen, ist also mit 8 Bit codiert. Damit hat man gewissermaßen eine Schwingungsform erzeugt, auf die eine Fourieranalyse angewandt werden kann.

Für die Fourieranalyse wurden sehr schnelle Algorithmen entwickelt. Gespeichert werden die Koeffizienten,

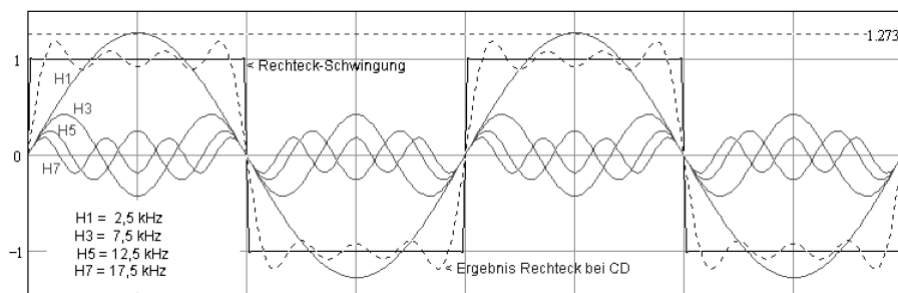


Bild 5: Jede periodische Schwingungsform lässt sich in eine Additionsreihe zerlegen, deren einzelne Glieder aus einfachen Sinusschwingungen bestehen. Das Beispiel zeigt eine Rechteck-Schwingung.

<http://www.omtec-audio.de/>

http://upload.wikimedia.org/wikipedia/de/3/37/Gibbsches_Ph%C3%A4nomen.png

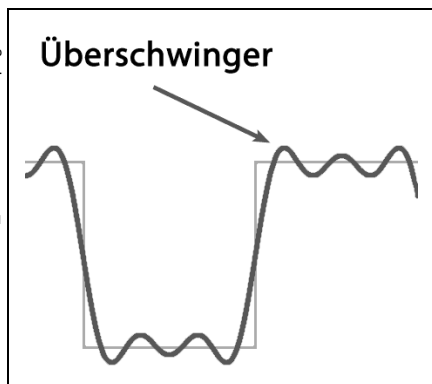


Bild 6: Hohe Frequenzanteile bei der Fourieranalyse des Signals sorgen für Überschwinger. Dadurch werden die Bildfehler beim JPEG-Format verursacht.

die sich aus der Fourieranalyse ergeben haben. Der Inhalt vieler Bilder besteht nur zu einem geringen Anteil aus scharfen Kanten; meist liegen Flächen vor. Eine Fläche besitzt nun einen geringen Anteil hoher Frequenzen, folglich müssen nur wenige Koeffizienten gespeichert werden. Das ist der Grund für die beeindruckenden Kompressionsraten, die bei JPEG erreicht werden können.

Die Betrachtung der scharfen Kanten führt uns zum Gibbs'schen Phänomen: Eine scharfe Kante (Rechteckkurve) ist durch einen hohen Frequenzanteil gekennzeichnet. Die Fourieranalyse kann solche Kanten nicht exakt darstellen; es kommt zu „Überschwingern“ (Bild 6). Und diese Überschwinger sind genau die „Verunreinigungen“, die beim JPEG-Bild zu beobachten sind! Unmittelbar an der Kante sind die Auslenkungen besonders groß, deshalb finden wir hier schwarze Pixel, die den Rand gewissermaßen nachzeichnen. Weiter vom Rand entfernt sind die Anteile hoher Frequenzen geringer, damit ist die Farbabweichung der Pixel nicht mehr so stark.

Methodische Hinweise

Mit diesem Experiment kann zum einen der Begriff Digitalisierung am Beispiel von Bilddateien deutlich gemacht werden. Vor- und auch Nachteile der Digitalisierung lassen sich leicht herausarbeiten.

Die Lehrkraft sollte nach der Erläuterung des Bitmap-Formats gemeinsam mit den Schülerinnen und Schülern die Bildgröße nach der angegebenen Formel berechnen und mit dem tatsächlichen Wert vergleichen. Damit ist die Basis für die Wertschätzung von Kompressionsverfahren gelegt, die bei nur geringen Einbußen an Bildqualität eine enorme Verkleinerung der Dateien und so z. B. eine schnelle Übertragung von Grafiken und Bildern im Internet ermöglichen.

Deutlich wird ebenfalls, dass nicht alle Bildfehler der Digitalisierung „in die Schuhe geschoben“ werden können. Das beobachtete Gibbs'sche Phänomen resultiert aus der im Kompressionsverfahren angewandten Fourieranalyse. Andere Kompressionsverfahren verwenden Algorithmen, die verlustfrei Bilder speichern können und dabei noch kleinere Dateien erzeugen. Ein Beispiel ist das Grafikformat PNG (*Portable Network Graphics*); ein Speichern der Bitmap-Datei in diesem Format und dem höheren Filtertyp reduziert die Dateigröße auf nur 4 kB!

Damit wird auch an diesem Beispiel wieder einmal deutlich, dass der Auswahl von problemadäquaten Algorithmen eine entscheidende Bedeutung zukommt!

Im Internet finden sich etliche JAVA-Programme, mit denen sich die Fourieranalyse simulieren lässt. Sie können das Verständnis der Fourieranalyse erleichtern und zeigen auch das Gibbs'sche Phänomen bei der Fourieranalyse einer Kante (Bild 6). Bezugsadressen für zwei solcher Programme finden sich weiter unten in den Internetquellen.

Im Anschluss an die Erläuterungen können die Schülerinnen und Schüler die mitgebrachten Fotos analysieren und auf ihnen das Gibbs'sche Phänomen entdecken.

Jürgen Müller
Berufsakademie Gera
Staatliche Studienakademie Thüringen
Weg der Freundschaft 4a
07546 Gera

E-Mail: juergen.mueller@ba-gera.de

Dieser Beitrag kann durch jedermann gemäß den Bestimmungen der „Lizenz für die freie Nutzung unveränderter Inhalte“ genutzt werden. Die Lizenzbedingungen können u. a. auf der Internetpräsenz des LOG IN Verlags (<http://www.log-in-verlag.de/zeitsch.html>) abgerufen werden (siehe auch S. 76 in diesem Heft).

Im LOG IN-Service (siehe S. 76) stehen die freie Bildbearbeitungssoftware PhotoFiltre, diverse Bilddateien sowie das JAVA-Applet von Falstad zur Visualisierung der Fourieranalyse zur Verfügung.

Literatur:

Meffert, B.; Hochmuth, O: Werkzeuge der Signalverarbeitung. München: Pearson-Studium, 2004.

Internet-Quellen [Stand: April 2007]

JAVA-Applets zur Visualisierung von Fourierreihen:
<http://www.mathe-online.at/galerie/fourier/fourier.html>
bzw.
<http://www.falstad.com/fourier/>

PhotoFiltre:
<http://www.photofiltre.com/>
bzw.
http://photofiltre.free.fr/frames_en.htm

Anzeige