Ernst Geretschläger **ernst.geretschlaeger@htl-steyr.ac.at**

Bildkomprimierung

- Mathematische / Fachliche Inhalte in Stichworten:

  Matrizenrechnung, diskrete Kosinustranformation

- Kurzzusammenfassung

  Bei der Bildspeicherung treten sehr rasch große Datenmengen auf. Zur Einsparung von Speicherplatz muss man Wege finden, wie man die Bildinformationen komprimiert abspeichert.
  Ein übliches Format ist das JPG- oder JPEG-Format. Es basiert auf der Idee der diskrekten Kosinustransformation. Für die Anwendung des JPG-Verfahrens wird ein Bild üblicherweise in 8x8 Matrizen von Pixelwerten zerlegt - unser Beispiel verfolgt nun, was mit einer dieser 8x8 Matrizen gemacht wird.

- Didaktische Überlegungen

  Angewandte Beispiele aus der Matrizenrechnung führen oft zu umfangreicher Rechnerei, daher ist auf diesem Gebiet der Einsatz des Computers oder eines modernen Taschenrechners sinnvoll.
  Für die Berechnung sollten grundlegende Kenntnisse über die Behandlung von Matrizen in Mathcad vorhanden sein - u. a. Einfügen einer Matrix, elementweises Erstellen einer Matrix über Matrizenindices, Transponieren und Matrizenmultiplikation.
  Zur Kontrolle des Verständnisses empfiehlt es sich, das Schema zur Datenkompression mit nur 2x2 Matrizen händisch durchführen zu lassen. Die technischen Details der Handhabung von Mathcad sollten jedenfalls schon beherrscht werden.

- Lehrplanbezug :

  Elektronik, Informatik, angewandte Mathematik

- Mathcad-Version:

  Mathcad 11

- Literaturangaben:

  http://www.binaryessence.de/dct/de000164.htm
  http://www.calsky.com/lexikon/de/txt/d/di/diskrete_kosinustransformation.php

  Linkangaben ohne Gewähr.

Ablauf:

1) Die Grauwerte eines Bildes werden in 8x8 Matrizen unterteilt. Die Grauwerte liegen zwischen 0 und 255. Das Beispiel wird nur für eine 8x8 Matrix durchgeführt.

2) Normierung: Von allen Grauwerten wird 128 subtrahiert.

3) Die diskrete Kosinustransformation (dct) wird für jede 8x8 Matrix durchgeführt. Es handelt sich dabei um eine Matrizenmultiplikation mit der vorgegebenen Matrix dct, deren Elemente nach untenstehender Formel (*) erstellt werden. Sie wandelt das Zeitsignal der Grauwerte in das Spektrum mit Grund- und Oberschwingungen um. Durch das Runden auf Einer werden die kleinen Amplituden der Oberschwingungen tlw. ausgelöscht.

4) Komprimierung durch elementweises Dividieren durch "Qualitätsfaktoren" und Runden des Ergebnisses auf Einer.

Damit die Matrizenindizeszählung - wie üblich - bei 1 beginnt setzt man $\mathrm{ORIGIN} := 1$.

Erstellen der Transformationsmatrix *dct*

Zeilen- bzw. Spaltenanzahl maximal $n := 8$, für den Taschenrechner reichen 4 bis 6. Anschließend werden die Matrixelemente nach folgender Vorschrift gebildet:

(*)

1. Zeile $\quad \mathrm{spalte} := 1..n \quad dct_{1,\mathrm{spalte}} := \frac{1}{\sqrt{n}}$

2. u. folg. Zeilen $\quad \mathrm{zeile} := 2..n \quad dct_{\mathrm{zeile},\mathrm{spalte}} := \sqrt{\frac{2}{n}}\cdot\cos\left[(2\cdot\mathrm{spalte}-1)\cdot(\mathrm{zeile}-1)\cdot\frac{\pi}{2\cdot n}\right]$

$$dct = \begin{pmatrix} 0.354 & 0.354 & 0.354 & 0.354 & 0.354 & 0.354 & 0.354 & 0.354 \\ 0.49 & 0.416 & 0.278 & 0.098 & -0.098 & -0.278 & -0.416 & -0.49 \\ 0.462 & 0.191 & -0.191 & -0.462 & -0.462 & -0.191 & 0.191 & 0.462 \\ 0.416 & -0.098 & -0.49 & -0.278 & 0.278 & 0.49 & 0.098 & -0.416 \\ 0.354 & -0.354 & -0.354 & 0.354 & 0.354 & -0.354 & -0.354 & 0.354 \\ 0.278 & -0.49 & 0.098 & 0.416 & -0.416 & -0.098 & 0.49 & -0.278 \\ 0.191 & -0.462 & 0.462 & -0.191 & -0.191 & 0.462 & -0.462 & 0.191 \\ 0.098 & -0.278 & 0.416 & -0.49 & 0.49 & -0.416 & 0.278 & -0.098 \end{pmatrix}$$

Eingabe oder Einlesen der Grauwerte

Die Bilddaten werden als 8x8 Matrix von Pixel - z. B. Grauwerte - händisch eingegeben oder aus einer Vorlage herausgelesen.

$$pixgrau := \begin{pmatrix} 85 & 78 & 78 & 77 & 85 & 78 & 85 & 93 \\ 143 & 144 & 151 & 151 & 143 & 170 & 183 & 181 \\ 153 & 151 & 162 & 166 & 162 & 151 & 126 & 117 \\ 143 & 144 & 133 & 130 & 143 & 163 & 159 & 175 \\ 123 & 112 & 116 & 130 & 143 & 147 & 162 & 189 \\ 133 & 151 & 162 & 166 & 170 & 178 & 166 & 128 \\ 162 & 168 & 166 & 159 & 135 & 104 & 93 & 98 \\ 154 & 155 & 153 & 144 & 126 & 106 & 118 & 133 \end{pmatrix}$$

Normierung der Grauwerte

128 wird elementweise subtrahiert und in neuer Matrix "grau" gespeichert

$zeile := 1 .. n$

$grau_{zeile,spalte} := pixgrau_{zeile,spalte} - 128$ Normierung der Pixelwerte

$$grau = \begin{pmatrix} -43 & -50 & -50 & -51 & -43 & -50 & -43 & -35 \\ 15 & 16 & 23 & 23 & 15 & 42 & 55 & 53 \\ 25 & 23 & 34 & 38 & 34 & 23 & -2 & -11 \\ 15 & 16 & 5 & 2 & 15 & 35 & 31 & 47 \\ -5 & -16 & -12 & 2 & 15 & 19 & 34 & 61 \\ 5 & 23 & 34 & 38 & 42 & 50 & 38 & 0 \\ 34 & 40 & 38 & 31 & 7 & -24 & -35 & -30 \\ 26 & 27 & 25 & 16 & -2 & -22 & -10 & 5 \end{pmatrix}$$

Diskrete Kosinustransformation

Nun wird die eigentliche Transformation $trans := dct \cdot grau \cdot dct^{T}$ mit anschließender Rundung $trans_{zeile,\,spalte} := rund\left(trans_{zeile,\,spalte}\right)$ ausgeführt.

Man beachte, dass ausser der Rundung noch kein Datenverlust stattgefunden hat.

$$trans = \begin{pmatrix} 83 & 1 & -5 & -9 & 3 & 0 & 2 & -1 \\ -52 & -60 & 14 & 16 & -4 & 2 & 7 & -1 \\ -98 & 63 & 3 & -20 & 4 & 5 & -5 & 5 \\ -65 & -37 & -7 & 8 & -5 & 3 & -5 & 4 \\ -92 & -45 & 48 & -6 & 18 & -10 & 1 & 3 \\ -66 & 63 & -15 & 4 & 3 & -11 & 0 & 0 \\ -23 & 15 & -37 & 14 & -7 & 1 & -2 & 4 \\ -54 & 30 & -8 & -9 & 25 & -2 & 1 & 4 \end{pmatrix}$$

Komprimierung - Erstellen der Qualitätsmatrix

Die Komprimierung entsteht durch elementweises Dividieren der oben erhaltenen Matrix durch die Qualitätsfaktoren. Dazu wird die Qualitätsmatrix "qual" erstellt. Die Qualität der Transformation kann durch den Faktor $qualitaet := 2$ (je höher desto schlechter) bestimmt werden. Mit Hilfe dieses Wertes wird die Qualitätsmatrix qual erzeugt: $qual_{zeile,\,spalte} := 1 + (zeile + spalte - 1) \cdot qualitaet$. Es werden dabei die rechts unten stehenden Elemente stärker reduziert - entsprechend die Reduktion der Oberschwingungen.

$$qual = \begin{pmatrix} 3 & 5 & 7 & 9 & 11 & 13 & 15 & 17 \\ 5 & 7 & 9 & 11 & 13 & 15 & 17 & 19 \\ 7 & 9 & 11 & 13 & 15 & 17 & 19 & 21 \\ 9 & 11 & 13 & 15 & 17 & 19 & 21 & 23 \\ 11 & 13 & 15 & 17 & 19 & 21 & 23 & 25 \\ 13 & 15 & 17 & 19 & 21 & 23 & 25 & 27 \\ 15 & 17 & 19 & 21 & 23 & 25 & 27 & 29 \\ 17 & 19 & 21 & 23 & 25 & 27 & 29 & 31 \end{pmatrix}$$

Komprimierung

Komprimierung durch elementweises Dividieren und Runden auf Einer:

$$komp_{zeile,\,spalte} := rund\left(\frac{trans_{zeile,\,spalte}}{qual_{zeile,\,spalte}}, 0\right)$$

$$komp = \begin{pmatrix} 28 & 0 & -1 & -1 & 0 & 0 & 0 & 0 \\ -10 & -9 & 2 & 1 & 0 & 0 & 0 & 0 \\ -14 & 7 & 0 & -2 & 0 & 0 & 0 & 0 \\ -7 & -3 & -1 & 1 & 0 & 0 & 0 & 0 \\ -8 & -3 & 3 & 0 & 1 & 0 & 0 & 0 \\ -5 & 4 & -1 & 0 & 0 & 0 & 0 & 0 \\ -2 & 1 & -2 & 1 & 0 & 0 & 0 & 0 \\ -3 & 2 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

Die kompirmierte Matrix enthält viele Nullen. Durch Umwandeln in eine Zahlenfolge erhält man bei geeigneter Nummerierung etliche Nullbytes.

Rücktransformation und Kontrolle der Qualität

Durch Rücktransformation kann man die Qualitätsverluste kontrollieren.

1. Schritt:
   Rückgängig machen der Division durch die Qualitätsfaktoren

$$rtrans_{zeile,\,spalte} := komp_{zeile,\,spalte} \cdot qual_{zeile,\,spalte}$$

$$rtrans = \begin{pmatrix} 84 & 0 & -7 & -9 & 0 & 0 & 0 & 0 \\ -50 & -63 & 18 & 11 & 0 & 0 & 0 & 0 \\ -98 & 63 & 0 & -26 & 0 & 0 & 0 & 0 \\ -63 & -33 & -13 & 15 & 0 & 0 & 0 & 0 \\ -88 & -39 & 45 & 0 & 19 & 0 & 0 & 0 \\ -65 & 60 & -17 & 0 & 0 & 0 & 0 & 0 \\ -30 & 17 & -38 & 21 & 0 & 0 & 0 & 0 \\ -51 & 38 & 0 & 0 & 25 & 0 & 0 & 0 \end{pmatrix}$$

2. Schritt:
Umkehrung der diskreten Kosinustransformation - man löst dazu die Matrizengleichung trans = dct.grau.dct^T durch Multiplikation mit den inversen Matrizen dct^{-1} von links und $(dct^T)^{-1}$ von rechts nach grau auf: $rgrau := dct^{-1} \cdot rtrans \cdot \left(dct^T\right)^{-1}$

$$rgrau = \begin{pmatrix} -41.715 & -49.316 & -50.574 & -44.296 & -42.936 & -46.975 & -44.51 & -36.53 \\ 5.371 & 21.559 & 27.334 & 18.243 & 18.821 & 37.849 & 56.066 & 61.731 \\ 26.355 & 24.925 & 28.006 & 33.837 & 31.997 & 18.438 & 1.45 & -8.972 \\ 10.767 & 11.41 & 8.541 & 6.48 & 14.615 & 30.677 & 42.354 & 45.445 \\ 6.925 & -12.472 & -16.499 & 2.536 & 17.749 & 22.073 & 35.195 & 54.762 \\ 1.112 & 18.261 & 31.011 & 36.561 & 47.432 & 52.471 & 30.278 & -1.48 \\ 31.447 & 40.855 & 44.783 & 30.746 & 3.306 & -20.75 & -30.537 & -30.506 \\ 24.009 & 27.473 & 27.67 & 16.112 & -4.909 & -18.478 & -12.569 & 0.986 \end{pmatrix}$$

3. Schritt: Normierung aufheben $rgrau_{zeile,\,spalte} := rund\left(rgrau_{zeile,\,spalte} + 128\right)$

$$rgrau = \begin{pmatrix} 86 & 79 & 77 & 84 & 85 & 81 & 83 & 91 \\ 133 & 150 & 155 & 146 & 147 & 166 & 184 & 190 \\ 154 & 153 & 156 & 162 & 160 & 146 & 129 & 119 \\ 139 & 139 & 137 & 134 & 143 & 159 & 170 & 173 \\ 135 & 116 & 112 & 131 & 146 & 150 & 163 & 183 \\ 129 & 146 & 159 & 165 & 175 & 180 & 158 & 127 \\ 159 & 169 & 173 & 159 & 131 & 107 & 97 & 97 \\ 152 & 155 & 156 & 144 & 123 & 110 & 115 & 129 \end{pmatrix}$$

Fehlerauswertung:

absoluter Fehler: $absfehl_{zeile,spalte} := \left|pixgrau_{zeile,spalte} - rgrau_{zeile,spalte}\right|$

Hinweis: Im Betrag dürfen nur Zahlen stehen, daher muss hier die Matrix elementweise aufgebaut werden.

$$absfehl = \begin{pmatrix} 1 & 1 & 1 & 7 & 0 & 3 & 2 & 2 \\ 10 & 6 & 4 & 5 & 4 & 4 & 1 & 9 \\ 1 & 2 & 6 & 4 & 2 & 5 & 3 & 2 \\ 4 & 5 & 4 & 4 & 0 & 4 & 11 & 2 \\ 12 & 4 & 4 & 1 & 3 & 3 & 1 & 6 \\ 4 & 5 & 3 & 1 & 5 & 2 & 8 & 1 \\ 3 & 1 & 7 & 0 & 4 & 3 & 4 & 1 \\ 2 & 0 & 3 & 0 & 3 & 4 & 3 & 4 \end{pmatrix}$$

Prozentueller Fehler - wenn alle Grauwerte ungleich 0 sind !!! - :

$$prozfehl_{zeile,spalte} := wenn\left[pixgrau_{zeile,spalte} = 0, \text{"n. def."}, rund\left[\left|\left(\frac{pixgrau_{zeile,spalte} - rgrau_{zeile,spalte}}{pixgrau_{zeile,spalte}}\right)\cdot 100\right|, 1\right]\right]$$

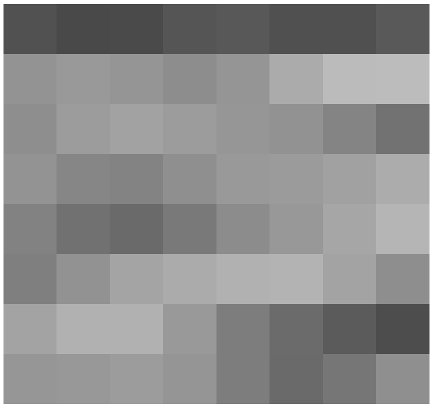
$$prozfehl = \begin{pmatrix} 1.2 & 1.3 & 1.3 & 9.1 & 0 & 3.8 & 2.4 & 2.2 \\ 7 & 4.2 & 2.6 & 3.3 & 2.8 & 2.4 & 0.5 & 5 \\ 0.7 & 1.3 & 3.7 & 2.4 & 1.2 & 3.3 & 2.4 & 1.7 \\ 2.8 & 3.5 & 3 & 3.1 & 0 & 2.5 & 6.9 & 1.1 \\ 9.8 & 3.6 & 3.4 & 0.8 & 2.1 & 2 & 0.6 & 3.2 \\ 3 & 3.3 & 1.9 & 0.6 & 2.9 & 1.1 & 4.8 & 0.8 \\ 1.9 & 0.6 & 4.2 & 0 & 3 & 2.9 & 4.3 & 1 \\ 1.3 & 0 & 2 & 0 & 2.4 & 3.8 & 2.5 & 3 \end{pmatrix}$$

Fehler liegt unter 10 %!

Bildvergleich:

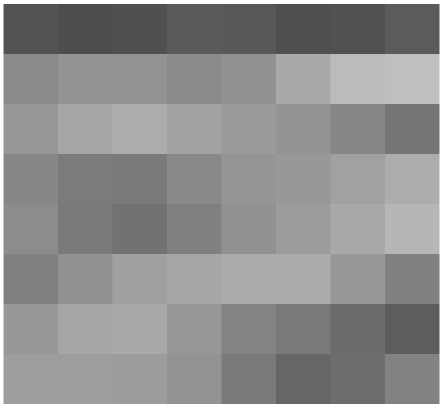
Ein Bildvergleich erstellt aus den Matrizen pixgrau und rgrau:
(Man erhält das Bild über "Einfügen" -> "Bild")

Original



pixgrau

Rücktransformierte

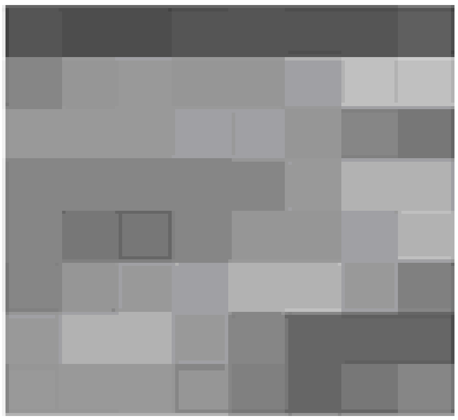


rgrau

Hinweis

Falls beim Bildvergleich nur kleine Bereiche zu sehen sind, diese Markieren und Aufziehen - funktioniert in Version 2000 Professional. Solllte das noch nicht funktionieren, dann im Bild "rechte Maustaste" -> Zoom und "An Fesnster anpassen"auswählen.

Man kann ansonsten versuchen die Bilder selbst zu erstellen: Menü Einfügen Bild und in den Platzhalter den Namen der verwendeten Matrix ("pixgrau" oder "rgrau") eingeben und anschließend aufziehen.
Falls dies alles nicht funktioniert, habe ich die beiden "Grauwertbilder" vergrößert als nicht veränderbare Bilder kopiert.



Hinweise zu weiteren Untersuchungen:

1.) Da das Beispiel als Anwendung der Matrizenrechnung konzipiert ist, wurde der theoretische Hintergrund über die Kosinustransformation nicht behandelt.
2.) Falls man zum Erstellen der Grauwertmuster zufallsgenerierte Matrizen verwendet, erreicht man wenig Nullen in der komprimierten Matrix. Dies hängt damit zusammen, dass zufällige Muster praktisch nicht durch einen Algorithmus komprimiert werden können - sie wären ja sonst nicht zufällig.
3.) Durch Verändern des Faktors *qualitaet* kann man die Güte der Kompression beeinflussen - oder man versucht andere "Qualitätsmatrizen" zu kreieren.
4.) Experimente mit "scharfen Kanten" - Bilder die scharfe Konturen haben.