

Elektronische Rasterverfahren

Problem:

Die APA Computer Outputprinter erlauben normalerweise nur eine 2-Dimensionale optische Dichte z.B. Drucken eines Tintenpunktes oder nicht-Drucken eines Tintenpunktes. Die Signalwerte von Bilder enthalten jedoch meist kontinuierlich variierende Tonwerte.

Idee:

Man nimmt ein Grauwertbild (amplitudenmoduliert Halbtonbild, das nur aus Schwarz-, Weiß- und Graudaten besteht), lässt es durch einen Rasteralgorithmus („halftoning“) bearbeiten und erhält so ein Binärbild (dichtemoduliert Bild, das nur schwarze und weiße Pixel enthalten).

Loesungen:

1. Photographisches Rasterverfahren

Wird in der Lithographie und im Lichtdruck verwendet.

Ein Halbtonnegativbild wird von einer stetigen Tonkopie (continuous-tone copy) angefertigt, indem ein Halbtonbildschirm (halftone screen) zwischen die Kameralinse und einen Hochkontrast-Photofilm gelegt wird (im Lichtstrahl).

Die Aufgabe des Halbtonbildschirmes und des Hochkontrast-Photofilmes ist, die stetigen Tonwerte des Originaltonbildes in Punkte mit gleicher Dichte aber unterschiedlicher Grösse, deren Mittelpunkte auf einem Gitter platziert werden, zu transformieren.

Dieses Verfahren wird meist im Offsetdruck verwendet. Bei herkömmlichen Tintenstrahldruckern wäre es jedoch technisch schwierig, solche verschiedenen grosse Farbfüllungen zu erzeugen.

2. Digitales Rasterverfahren

Algorithmen:

Constant-level thresholding:

Ein konstanter Grenzwert $t(x)$ wird bestimmt, nur diejenigen Pixel des Bildes werden gedruckt, deren Signalwerte $q(x)$ sich oberhalb dieses Wertes befinden ($q(x) > t(x)$).

Variable-level thresholding:

Ein variabler Grenzwert (Dither pattern) $t[\text{mod}(x,i)]$ wird bestimmt, mit welchem der Signalwert $q(x)$ jedes Pixel verglichen wird. Ist der Signalwert oberhalb des Grenzwertes, wird der Pixel gedruckt, ansonsten nicht ($q(x) > t(\text{mod}(x,i))$).

Schwellwertmatrizen:

Dieses Verfahren kann man natürlich auch 2-Dimensional anwenden. In diesem Fall hat man eine Matrix welche mit Schwellwerten gefüllt ist. Die jeweiligen Signalwerte der einzelnen Pixel des Bildes $q(x,y)$ werden mit den korrespondierenden Werten in der Schwellwertmatrix $t[\text{mod}(x,i), \text{mod}(y,j)]$ verglichen und je nachdem gedruckt oder nicht ($q(x,y) > t[\text{mod}(x,i), \text{mod}(y,j)]$).

1. Dispersed dot-pattern generation:

i -ter und $(i+1)$ -ter Schwellwert der Matrix liegen so weit wie möglich auseinander. Unter idealen Bedingungen, wird ein Schachbrett-ähnliches Muster generiert. Die Bedingungen sind jedoch meistens nicht ideal, da die meisten Drucker keine quadratischen, sondern nur runde Flächen drucken können.

2. Clustered dot-pattern generation:

i -ter und $(i+1)$ -ter Schwellwert der Matrix liegen so nahe wie möglich beisammen. Mittels der Überlagerung von Punkten, können rechteck-ähnliche Formen approximiert werden. Dies führt zu besseren Grautonwertwerten als beim Dispersed dot-pattern Verfahren. Ein guter Ansatz für das clustered dot pattern Verfahren ist das Konzept des *Super-Circle* in welchem die dominanten weissen und schwarzen Muster die gleiche Form haben. Die dominanten Muster im dunklen Gebiet des Graustufenbildes sind kreisförmige weisse Muster,

welche ihre Form mit der Veränderung der Farb gegen die Mitteltöne hin langsam in ein Quadrat ändern. Ähnlich ist es auf der anderen Seite. Die dominanten Muster im weissen Gebiet der Graustufenbildes sind kreisförmige schwarze Muster, welche ihre Form mit der Veränderung der Farbe gegen die Mitteltöne hin langsam in ein Quadrat ändern. Dies führt zu einem sanften Übergang eines Kreises zu einem Quadrat und wieder zurück zu einem Kreis. Das Clustering von Schwellwerten wird aufgrund der folgenden Funktion berechnet.

$$X^{2n} + Y^{2n} = r^{2n} \text{ für } 1 \leq n < \infty.$$

Weiter gilt:

$$A = 4 r^2 [\Gamma(1/2n + 1)]^2 / \Gamma(1/n + 1), \text{ (}\Gamma \text{ ist die Gammafunktion).}$$

n beschreibt die Form der Körpers und zusammen mit r dessen Fläche A.

2Q ist die Anzahl der Superkreise und X und Y die Koordinaten.

Für n = 1 gilt die normale Kreisfläche πr^2 .

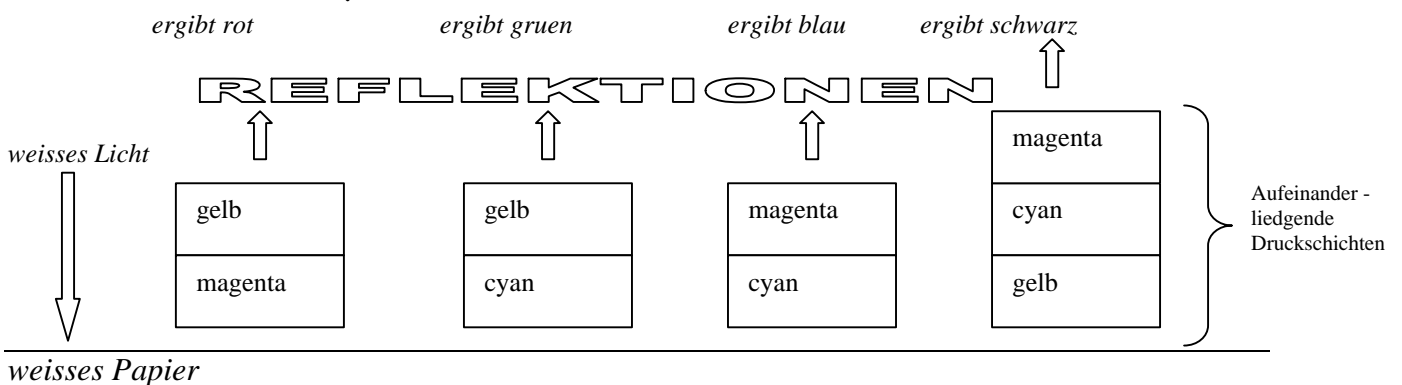
Für n gegen ∞ gilt die Rechteckfläche $4 r^2$

1. r als Funktion der Fläche A für Rechteck und Kreis bestimmen
2. Für die Q ersten Superkreise, reicht ein Kreisdesign zur Approximation aus.
3. Für die Q oberen Superkreise, wird eine Menge von geeigneten ‚r‘ gewählt und das Flächenprofil angepasst, indem man ‚n‘ von 1 (Kreis) bis ∞ (Rechteck) variiert und in die folgende Formel einsetzt.

$$r_{\text{chosen}}(A) = r_{\text{circle}}(A) - ((A-Q)/Q)[r_{\text{circle}}(A) - r_{\text{square}}(A)]$$

Farbige Bilder

Zuerst wird auf dem Originalbild ein Bandpass-Filter angesetzt um es in seine additiven Elementarfarben ‚grün, rot, blau‘ zu zersetzen. Die Farben werden dann abgeschirmt und für ihre Rekonstruktion auf weissem Papier deren subtraktive Elementarfarben ‚magenta (absorbiert gruen), gelb (absorbiert blau) und cyan (absorbiert rot)‘ verwendet.



Dies kann zu einem falschen Resultat führen, falls ein falscher Filter, oder ein falsches Papier verwendet wird, oder wenn die Pigmente in den Farben falsch sind.

Da schwarz beinahe nicht erzeugbar ist, fügt man es als vierte Farbe hinzu -> Aus dem dreifarben Druck entsteht ein vierfarben Druck. Wegen der Granularität können wir die Farben nicht genau aufeinander drucken, d.h. die einzelnen Schichten werden unter verschiedenen Winkeln (siehe Tabelle) aufeinander gedruckt. Wir produzieren also absichtlich Interferenzmuster hoher Frequenzen.

	3-farben-druck		4-farben-druck	
Gelb	90	grad	90	grad
Magenta	30	grad	15	grad
Cyan	60	grad	75	grad
Schwarz -			45	grad

Error-diffusion:

Wird im 2-dim. Raum angewendet, um die unerwünschte einheitliche Spektraldichteverteilung, welche bei pseudo-zufälligen Schwellwerten auftritt zu meiden. Der Fehlerwert $e(x)$ wird durch die Nachbarschaft von x und nicht nur durch dessen Vorgänger wie dies beim 'error-carry' im 1-dimensionalen Raum der Fall ist, bestimmt. Natürlich benötigt 'error-diffusion' mehr Prozessorleistung und mehr Speicherkapazität als dies beim 'error-carry' der Fall ist, dafür hat es jedoch den Vorteil, die räumliche Auflösung - (spatial-resolution) und Farbechtheitseinschränkungen (tone-fidelity) weitgehend zu beseitigen. Es gibt zwei verschiedene Strategien für den Algorithmus.

Strategie I

Teile des Fehlerwertes $e(x,y)$ jedes Pels an der Position (x,y) werden den anliegenden, digitalisierten, stetigen Tonsignalen übergeben ($q'(x,y) = q(x,y) + e(x,y)$) und dann mit dem Schwellwert verglichen

Strategie II

Ein gewichteter Durchschnitt im voraus berechneter Fehlerwerte der Nachbarschaft des zu druckenden Punktes, wird zum stetigen Tonsignalwert $q(x,y)$ an der Position (x,y) addiert und dann mit dem Schwellwert verglichen.

1. $q'(x,y) = q(x,y) + \sum_{i,j} (w(i,j)e(x+i,y+j)) / \sum_{i,j} w(i,j)$
2. $q'(x,y)$ mit dem Schwellwert $t(x,y)$ ($= Q/2$) vergleichen
 - a. if($q'(x,y) > t(x,y)$) then $h(x,y) = Q$ (schwarz) else $h(x,y) = 0$ (weiss)
3. $e(x,y) = q'(x,y) - h(x,y)$ wird in ein $S \times M$ Fehlerbuffermemory geschrieben um ihn bei einem späteren Algorithmdurchlauf wieder zu gebrauchen.

$q'(x,y)$:	korrigierter,digitaler,stetiger Tonwert
$q(x,y)$:	digitaler,stetiger Tonwert
$w(i,j)$:	Fehlerfilter Inputantwort über (i,j)
$e(x+i,y+j)$:	Fehlerwerte an der Position $(x+i,y+j)$
$t(x,y)$:	Schwellwert
$h(x,y)$:	Bi-level halbtone Representation

Constant-level-thresholding und 2-dimensionales Error-diffusion generieren eine deterministische Anordnung ungeordneter Punktmuster ('dot-patterns'). Der Grad der 'Unordnung' hängt von der Art des Fehlerfilters ab.

1.Erweiterung -> look-ahead-filter

Um die Bildung von Artefakten zu unterdrücken, werden constant-level-thresholding und 2-dimensionales Error-diffusion durch einen sogenannten 'look-ahead-filter' erweitert.

$$q'(x,y) = q(x,y) + [\sum_{i,j} (w(i,j)e(x+i,y+j)) / \sum_{i,j} w(i,j)] + [\sum_{k,l} (z(k,l)q(x+k,y+l)) / \sum_{k,l} z(k,l)]$$

Die Schritte 2. und 3. sind dann wieder gleich.
 $z(x,y)$: look-ahead-filter Inputantwort über (k,l)

Dieses Verfahren liefert noch nicht genügend gute Resultate für 'hardcopy' Reproduktionen auf nicht-idealen Druckern.

2.Erweiterung -> dot-size-correction

Dot-size-algorithm berechnen die Fläche die ein neu gedruckter Punkt neu abdeckt (d.h. vor dem Drucken dieses Punktes war diese Fläche noch weiss).

$$I(x,y) = A_{\text{effektiv}}(x,y) + A_{\text{pel}} = A_{\text{Punkt}}(x,y) - A_{\text{überlappung}}(x,y) + A_{\text{pel}} = p R^2 - A_{\text{überlappung}}(x,y) + A_{\text{pel}}$$

Alle möglichen $I(x,y)$ müssen nach jedem Druck eines Punktes berechnet und in einer Druckerspezifischen look-up-tabelle gespeichert werden (dies ergibt 2^L Werte: L = Anzahl adjazenter Druckpositionen).

Die Linearisierung hat grosse Vorteile bei Rasterdiffusionsverfahren für den Farbdruk, da er genaue Angaben über Mengen von CMYK-Werten liefert. Somit entstehen bei den Farbbildern keine 'Moirés'.

Ein Tool, das all diese Prozeduren beinhaltet ist das MECCA (multiple error correction computation algorithm).