

Rastertechnik

Vierfarbrasterung – die Sache mit Rosetten, Farbdrift und Moiré

Im dritten Teil seiner Abhandlung über Rastertechnik (Teil 1: DD 21–22/92, w70; Teil 2: DD 43/92, w12) geht Dipl.-Ing. (FH) Christian Greim auf Möglichkeiten zur moiréfreien Reproduktion von Vorlagen ein. Anschaulich beschreibt er Superzellen, Zufallsraster und stellt ein neues wellenüberlagertes Raster vor.

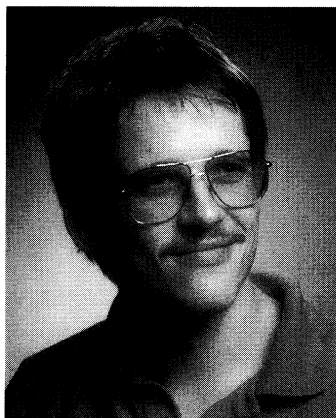
Wer diesen dritten Teil verstehen will, sollte entweder schon viel von *PostScript* und Rasterung wissen oder die beiden vorangegangenen Artikel lesen. Von meiner ursprünglichen Diplomarbeit, auf der das ganze fußt, blieb im Laufe der Zeit eigentlich nicht mehr viel übrig. Dieser Teil stellt noch keine fertigen Ergebnisse vor, der Artikel ist mehr ein Bastelbuch für Leute geworden, die neue Raster entwickeln oder die Gedanken hinter den einzelnen Rastern besser verstehen wollen.

Der Freiheitsgrad vom Quadrat zum Rechteck

Zwei Grundthesen liegen den weiteren Überlegungen zu Grunde: Erstens sind die bisherigen digitalen Raster nichts weiter als die digitale Form ihrer photographischen Urahnen (mit Ausnahme der frequenzmodulierten Raster); zweitens ist ein Übereinanderdruck von vier Farben nicht schon deshalb optimal, weil seine Einzelkomponenten das Optimum für einen einfarbigen Rasterdruck sind.

Die eigentlichen Interferenzprobleme treten nur auf, wenn zwei gleiche Rasterweiten unter einem kleinen Winkel überlagert werden. Wie im ersten Teil des Artikels beschrieben, kann man ein normales Raster auch als Überlagerung zweier Linienraster im 90°-Winkel verstehen, wobei gleichsam nur die Kreuzungspunkte vorhanden sind. Wer sagt aber, daß diese beiden Linienraster dieselbe Rasterweite haben müssen, wie das bisher ausschließlich der Fall ist? Bisher liegen jeweils vier benachbarte Punkte im Quadrat zueinander.

Man kann diese Punkte aber auch im Rechteck zueinander anordnen. Dadurch kann man vermeiden, daß zwei Grundraster mit gleicher Rasterweite in einem Winkel von 30° oder kleiner überlagert werden. Zur Verdeutlichung des eben Gesagten kann vielleicht die Abbildung 5 aus dem ersten Teil dieses Artikels dienen. Die einzelnen Rechteckraster müssen dann aber unter 60° verwindelt werden. Der Vorteil an der Sache ist, daß eine kleinere Rosette entsteht als beim herkömmlichen Raster.



Dipl.-Ing. (FH) C. Greim

Man vergleiche dazu die Abbildungen 16 und 17. Die Abbildung 16 ist ein ganz normales Raster. Abbildung 17 ist ein Rechteckraster, wobei eigentlich genauso viele Punkte pro Fläche vorhanden sein sollten, wie bei einem normalen Raster. Dabei sind aber leider die Tücken von *PostScript* voll zum Tragen gekommen. Der Belichter hat nämlich nur bei den Rasterwinkeln 0° und 45° die gewünschte Rasterweite beibehalten. Bei den anderen hat er leider die Rasterweite verwendet, die er speichermäßig gerade noch verkraften konnte.

Es entsteht zwar eine viel kleinere Rosette und das Raster ist gegen Winkelungsfehler viel weniger anfällig als ein normales Raster. Es kommt aber zu schwach sichtbaren streifigen Erscheinungen. Diese könnten allerdings von ungenauer Winkelung oder vom Belichter her kommen.

Wer das Rechteckraster selber ausprobieren will, kann gerne die *Setscreen*-Funktion eines *PostScript*-Codes entsprechend Abbildung 20, »%Rasterung im Verhältnis 2 zu 3, rationale Winkel, keine Welle« eingeben. Dabei verwende ich allerdings eine *Art Superzelle* mit jeweils 6 Punkten pro Zelle... ertappt!... Sie wissen nicht was eine Superzelle ist. Dazu müssen Sie jetzt entweder den zweiten Teil dieses Artikels im Kopf haben oder nachlesen. Die stark vergrößerte Zelle kann man in Abbildung 21 sehen. Die Superzelle hat die Ausdehnung von zwei Punkten in die eine und drei Punkten in die andere Richtung.

Also müssen Sie erst die Rasterweite berechnen, die Sie für *Setscreen* brauchen, damit Sie eine vorgegebene Rasterweite tatsächlich erhalten. Sie haben also in einer Zelle 6 Punkte. Wenn Sie die Wurzel aus 6 ziehen, dann haben Sie einen Faktor, der die Linienzahl je Rasterzelle angibt. Diesen Faktor müssen Sie je nachdem mit der gewünschten Rasterweite verrechnen.

Die neue alte Welle

Um nun die auftretenden streifigen Erscheinungen zu mindern, kam ich auf die Idee, einfach Wellenstrukturen zu überlagern. Auf diese Idee brachte mich ein altes Buch mit Hirnholzstichen, die aus

einem Linienraster in Wellenform aufgebaut sind. Bei meiner Wiederentdeckung dieses Prinzips wird die gesamte Superzelle mit einer Sinuskurve überlagert. Zunächst wurde der Sinus nur in Richtung der kurzen Rasterweite überlagert. Das Ergebnis war schon besser, wie Sie in Abbildung 18 sehen.

Die Abbildung 20 »Rasterung... mit Welle in kurzer Richtung, Phase 0,1, Amplitude 0,1, zeigt den Code für diese verbesserte Version. Dabei müssen Sie aber jeweils »/SP3« gegen »/SP5« in der »setcolorscreen«-Zeile ändern.

Die Vergrößerung dieser Zelle sehen Sie in Abbildung 22. Das spannende an der Sache ist, daß Welle nicht gleich Welle ist. Es muß bestimmt werden, wie groß die Amplitude der Welle ist. Dann muß auch noch festgelegt werden, wo der Nullpunkt der Welle im Bezug zur Superzelle steht. Eleganter spricht man dabei von der Phase der Welle.

Eigentlich könnte man auch noch die Form der Welle variieren. Da die Möglichkeiten dadurch ins unendliche wachsen, habe ich mich für die Otto-Normal-Welle, nämlich die Sinuswelle, entschieden. Da das Besere der Feind des Guten ist, habe ich auch in der anderen Richtung noch einmal eine Welle überlagert. Das Ergebnis ist in Abbildung 19 zu sehen.

Den Quellcode für die *PostScript Setscreen*-Funktion (siehe Teil 2) kann man in Abbildung 20 (»%Rasterung... mit Doppelwelle...«) sehen, dabei muß auch wieder die »setcolorscreen«-Zeile geändert werden.

Viele neue Möglichkeiten

Abbildung 23 zeigt die vergrößerte Zelle. Auch hier kann man die Amplitude, die Phase und die Form variieren. Das heißt, bei einer Farbe kann man allein über die Welle sechs Parameter variieren.

Durch die Vierfarbigkeit ergeben sich also 24 Parameter, die variiert werden können. Sollte jemand diese Raster tatsächlich kommerziell verwenden wollen, bitte ich ihn allerdings, sich vorher mit mir in Verbindung zu setzen.

Rückwirkend geht das natürlich nicht nur beim rechteckigen Raster, sondern die Welligkeit kann man auch in ein ganz normales Raster einbauen. Die übrigen Parameter wie Rasterweite und Rasterwinkel bleiben natürlich weiter variierbar. Damit läßt sich bei einem normalen Raster die Zahl der veränderlichen Parameter von zwei auf acht steigern, und das bei vier Farben macht also immerhin die stattliche Anzahl von 32 Parametern. Das birgt natürlich gigantisch viele Möglichkeiten in sich. Hier allein gibt es schon genügend Möglichkeiten, neue intelligenter Raster zu entwickeln.

Zufällig gibt es noch etwas Besseres

So lange man noch an absolut regelmäßigen Strukturen bei der Rasterung festhält, gibt es immer irgendwo Interferenzerscheinungen, sprich: etwas Ähnliches wie ein Moiré. Am besten ist da immer noch ein wirkliches *Zufallsraster*. In letzter Zeit gibt es darüber immer mehr zu lesen. So zufällig darf ein Zufallsraster nämlich auch nicht sein. Die Rasterpunkte sollen gleichmäßig unregelmäßig verteilt liegen. Dazu muß man die Bereiche möglichst vermeiden, wo Punkte aneinanderstoßen oder sehr eng beieinander liegen. Man spricht dabei vom Herausfiltern der hohen Frequenzen. Das ist in der Fläche gar keine so einfache Sache.

Das andere Problem ist, daß ein Zufallsraster auf PostScript-Belichtern von PostScript her eigentlich nicht machbar ist. PostScript hat zwar einen Zufallszahlengenerator eingebaut. Es ist auch überhaupt kein Problem, eine Setscreen-Funktion für einen Zufallsraster zu schreiben. Die Funktion ist sogar einfacher

```
%Zusatzbefehle
/BD (bind def)bind def
/ED (exch def)BD

%normale Rasterung
/SP1(abs /Y1 ED abs /X1 ED X1 Y1 add 1 ge (1 X1 sub dup mul 1 Y1 sub dup mul
add -1 add) (X1 dup mul Y1 dup mul add 1 exch sub) ifelse)BD
49 135 /SP1 load 49 15 /SP1 load 49 0 /SP1 load 49 75 /SP1 load setcolorscreen

%Rasterung im Verhältnis 2 zu 3, rationale Winkel, keine Welle
/SP3(1 add 1.4999 mul dup floor sub 2 mul 1 sub /Y1 ED 1 add 0.9999 mul
dup floor sub 2 mul 1 sub /X1 ED X1 Y1 add 1 ge
(1 X1 sub dup mul 1 Y1 sub dup mul add -1 add)
(X1 dup mul Y1 dup mul add 1 exch sub) ifelse)BD

%Befehl zum Einbau in das Programm bei einem Farbdrucker
20 127.62 /SP3 load 20 15 /SP3 load 20 0 /SP3 load 20 71.31 /SP3 load
setcolorscreen

%Rasterung im Verhältnis 2 zu 3, rationale Winkel, mit Welle in kurzer
%Richtung, Phase 0,1, Amplitude 0,1
/SP5(/Y1 ED /X1 ED Y1 0.1 add 180 mul sin 0.1 mul X1 add dup dup truncate
sub exch truncate abs mul add /X1 ED X1 Y1 1 add 1.4999 mul dup floor sub 2
mul 1 sub /Y1 ED 1 add 0.9999 mul dup floor sub 2 mul 1 sub /X1 ED X1 Y1 add
1 ge
(1 X1 sub dup mul 1 Y1 sub dup mul add -1 add)
(X1 dup mul Y1 dup mul add 1 exch sub) ifelse)BD

%Rasterung im Verhältnis 2 zu 3, rationale Winkel, mit Doppelwelle.
%Phase1: 0,1, Amplitude1: 0,1, Phase2: 0,5, Amplitude2: 0,1
/SP6(/Y1 ED /X1 ED Y1 0.1 add 180 mul sin 0.1 mul X1 add dup dup dup
truncate sub exch truncate abs mul add /X1 ED X1 0.5 add 180 mul sin 0.1 mul
Y1 add dup dup dup truncate sub exch truncate abs mul add /Y1 ED X1 Y1 1 add
1.4999 mul dup floor sub 2 mul 1 sub /Y1 ED 1 add 0.9999 mul dup floor sub 2
mul 1 sub /X1 ED X1 Y1 add 1 ge
(1 X1 sub dup mul 1 Y1 sub dup mul add -1 add)
(X1 dup mul Y1 dup mul add 1 exch sub) ifelse)BD
```

Abbildung 20: Setscreen-Funktion des PostScript-Codes

als die Standardpunktform. Sie lautet: pop pop rand 255 div and 255 div. Die einzelnen Befehle möchte ich hier nicht erläutern, das würde wirklich zu weit führen. Beim Betrachten der Abbildung 23 sieht aber vielleicht schon mancher, warum diese Funktion leider doch nicht funktioniert.

Das Problem dabei ist der Rasterpunkt. Im Prinzip handelt es sich um einen Rasterpunkt wie jeden anderen und genauso wird er immer wiederholt, was sich als störendes Muster bemerkbar macht. Es werden nämlich nicht immer neue Dots (Belichterpunkte) angesprochen, sondern bei dunkleren Graustufen kommen nur neue hinzu, was bei ähnlichen Grauwerten eigentlich eine ständige Wiederholung darstellt. Das ist der Nachteil daran,

daß bei üblichen RIPs die »threshold matrix« (siehe Teil 2 des Artikels) nur einmal berechnet und dann immer wiederholt wird. Der Effekt stört um so weniger, je größer die Rasterzelle ist. Am besten wären so etwa 5 cm Kantenlänge, aber das schafft leider kaum ein RIP. Bei größtmöglicher Kantenlänge und verschiedenen Rasterwinkeln läßt sich der Effekt etwas verringern. Bei dem Ganzen sollte man aber den ersten Effekt nicht vergessen. Mit dieser Einfachversion eines Zufallsrasters sind die Punkte nämlich noch lange nicht gleichmäßig verteilt. Mit etwas Mathematik läßt sich das aber beheben.

Wo ein Wille ist...

Man kann in PostScript auch die Befehle umgehen und eigene

Befehle programmieren. Damit kann auch der Halbtongenerator umgangen werden. Da die Befehle mühsam interpretiert werden, kann man dabei nicht nur die berühmte Kaffeepause machen, sondern man sollte den Rechner je nach Motiv einige Stunden in Ruhe lassen. Ich habe ein Zufallsraster einmal auf diese Weise für einen NeXT-Computer programmiert. Da das Programm leider sehr langsam und auf anderen PostScript-Interpretern kaum lauffähig ist, habe ich es nicht mit an den Verlag gegeben. Wer sich davon nicht abschrecken läßt, kann den Quellcode von mir direkt haben. Die Abbildung 25 zeigt jedenfalls, was man mit so einem Zufallsraster aus einem Laserdrucker herausholen kann. Es entspricht natürlich dem bekannten Runzelkornraster ohne eine Filterung der hohen Frequenzen.

Da gibt es doch aber irgendwelche Typen, die trotzdem einen Zufallsraster für PostScript haben. Das Ganze heißt mit Firmennamen *CrystalRaster* und hier sind die oben beschriebenen Probleme gelöst. So weit ich die Berichte dieser Firma verstanden habe, wird der PostScript-Code zuerst normal mittels eines Software-RIPs interpretiert. Danach hat man ja auch erst einmal einen Datenbestand. Diesen kann man dann mathematisch recht einfach so umstricken, daß daraus ein Zufallsraster wird, nur man braucht eben einen relativ langwierigen Rechenprozeß und zum Vor-ripen einen eigenen Rechner, falls man nicht schon ein Software-Rip hat. Insbesondere von einem Software-Rip namens *Freedom of Press* weiß ich, daß er nicht (wie in Teil 2 beschrieben), nur in ein hoch aufgelöstes Schwarzweißbild aufrastern kann, sondern er kann die Graustufen auch als

DAS ORIGINAL...
just do it...
HOTLINE zum Nulltarif
Tel: 0130-869 998

JUST
NORMLICHT

wozu warten? - Sichern Sie Ihr Qualitätsniveau und optimieren Sie Ihre Präsentationen mit JUST Normlicht 5000 Kelvin Professionell, - vom Dia bis zum fertigen Druck, - vom Einzelgerät bis zur Komplettausstattung.

"Color-Match Compact"
zur Farbabstimmung bei DTP, DTR und EBV
Aktionspreis DM 2100,-
(bei Kauf zzgl. MwSt.)

JUST NORMLICHT VERTRIEBS GMBH - P.O. BOX 20 - WEST GERMANY - WEILHEIM/TECK - TEL.: 07023 / 2060 - FAX: 07023 / 4754

Graustufen beibehalten. Dann ist das nachträgliche »Umrastern« auf ein Zufallsraster natürlich besonders einfach. Von der Firma BSL in Itzehoe wurde auf Anregung der Fachhochschule für Druck auf ähnliche Weise mittels Lasergravur sogar ein Zufallsraster für den Flexodruck hergestellt und an der Fachhochschule für Druck angedruckt. Obwohl noch keine Gradationskennlinie bekannt war, waren die ersten Versuche recht vielversprechend.

Wieviel Zufall braucht das Raster?

Beim Offsetdruck ist die Sache klar. Da es den CrystalRaster gibt und das Prinzip nicht all zu kompliziert ist, wird es bald noch mehr in dieser Richtung geben. Das Raster läßt sich anscheinend sehr gut verdrucken und reagiert sogar auf Überfärbungen weniger kritisch als ein herkömmliches Raster. Nur bei der Formherstellung muß man sich mehr Mühe geben als bisher.

Nachdem nun aber auch für den Flexodruck ein Zufallsraster erstmalig angedruckt war und bereits heftig Versuche mit Photopolymerplatten laufen, könnte dies auch ein Weg für den Flexodruck sein. Zunächst bleibe ich aber bei der Einschätzung, daß sich ein gutes Zufallsraster im Siebdruck kaum verwirklichen läßt.

Was also sollen nun die armen Anwender dieses Druckverfahrens tun? Einfach eine Kreuzung. Das hört sich jetzt natürlich nach einem blöden Spruch an. Aber wer sagt denn,

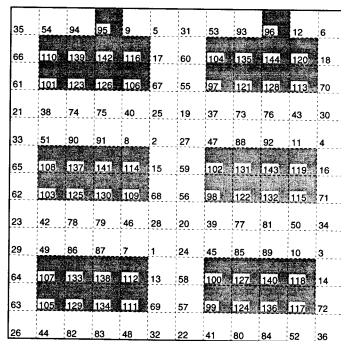


Abbildung 21: Simulation der Spotfunktion »[SP3]«.

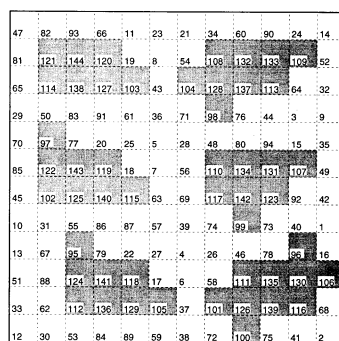


Abbildung 23: Simulation der Spotfunktion »[SP6]«.

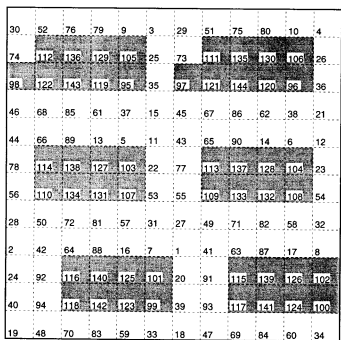


Abbildung 22: Die Simulation der Spotfunktion »[SP5]«.

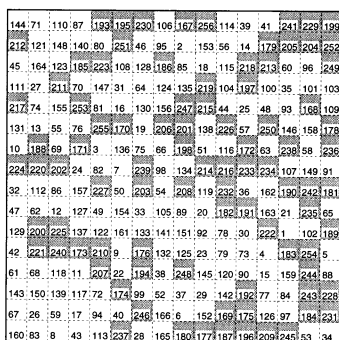


Abbildung 24: Simulation der Spotfunktion »[pop pop rand 255 and 255 div]«.

daß bei einem normalen Raster die Punkte immer in einer bestimmten Position stehen müssen? Man könnte doch bei einem normalen Raster die Punkte immer nur zufallsgesteuert ein bißchen aus ihrer ursprünglichen Lage stoßen, ebenso die negativen Punkte. Dadurch wäre es dann auch nicht mehr so auffällig, wenn man bei sehr hellen Tönen einfach weniger Rasterpunkte

schröbe. Eine sehr alte Idee, die wahre Wundervollbringen könnte, weil man hier große Probleme mit dem kleinsten noch druckenden Punkt hat. Wenn man diese Idee allerdings bei einem normalen Raster anwendet, entstehen sehr unschöne Strukturen. Mit solch einem Raster hätte man auch das ganze Grundproblem der Rosette zwar noch nicht eliminiert, aber doch so entschärft, daß nichts mehr passieren könnte. Außerdem wären die großen Probleme im Siebdruck mit Siebstrukturmoirés einfach erledigt. Der Haken an der Sache ist nur wie beim Zufallsraster, daß sich die Sache nicht so ohne weiteres in PostScript verwirklichen läßt. Auch hier müßte wie beim CrystalRaster gleichsam ein »Nachbrenner« eingebaut werden. Und diese Möglichkeiten hatte ich in meiner Zeit an der Fachhochschule leider nicht. Ich habe zwar mit Hilfe einer Art Superzellentechnik versucht, dieses Problem zu umgehen, bin aber genau an den Punkten gescheitert, die ich vorher schon beim Zufallsraster erwähnt habe.

Am Ende des Anfangs

Dies ist nun also das Ende dieses Artikels. Viele, die fix und fertige Lösungen erwartet haben, werden enttäuscht sein. Zu meiner Verteidigung sei gesagt, daß das Thema sehr komplex ist. Ich hoffe aber, daß dieser Artikel mit seinen drei Teilen nicht einfach beiseite gelegt wird und so getan wird, als sei das bisher an Rastern Erreichte das Optimum. Wer das sagt, macht sich und anderen nur etwas vor. Die Entwicklung einiger Anbieter hin zum Zufallsraster macht mir da aber große Hoffnung. Das Thema ist mit dem Wenigen, was ich in diesem dreiteiligen Artikel erwähnt habe, nur angerissen. Da gibt es zum Beispiel einen Mathematiker namens PENROSE, der sich mit Problemen der Parkettierung beschäftigt und dabei ein Muster gefunden hat, das durch keine Abbildung mit sich selbst zur Deckung gebracht werden kann. Auf dieser Grundlage hätte ein Raster natürlich auch viele Chancen... aber vielleicht hat jemand anderes viel Zeit und Ideen, um hier weiter zu machen.

SCHLÜSSELWÖRTER:
Interferenzerscheinung
Moiré
PostScript
Superzelle
Zufallsraster

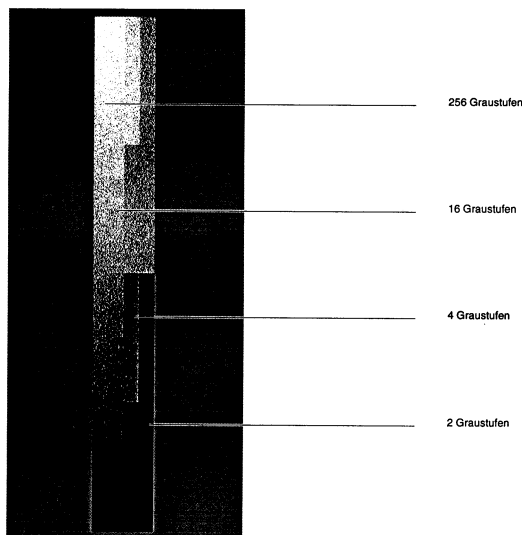


Abbildung 25: Das Zufallsraster des Laserdruckers entspricht dem eines Runzelkornrasters ohne Filterung der hohen Frequenzen.

DD-Jahrbücher 92/93

- Reproduktion und Druckformherstellung,
- Satz & DTP,
- Drucktechnik und Weiterverarbeitung

liefern Ihnen ein Konzentrat der wichtigsten Entwicklungen, Trends und Ereignisse des ganzen Jahres! DD-Jahrbücher sind einzigartige Fach- und Marktdokumentationen, die jede Fach- und Führungskraft besitzen sollte.