

## Aufgabe 5

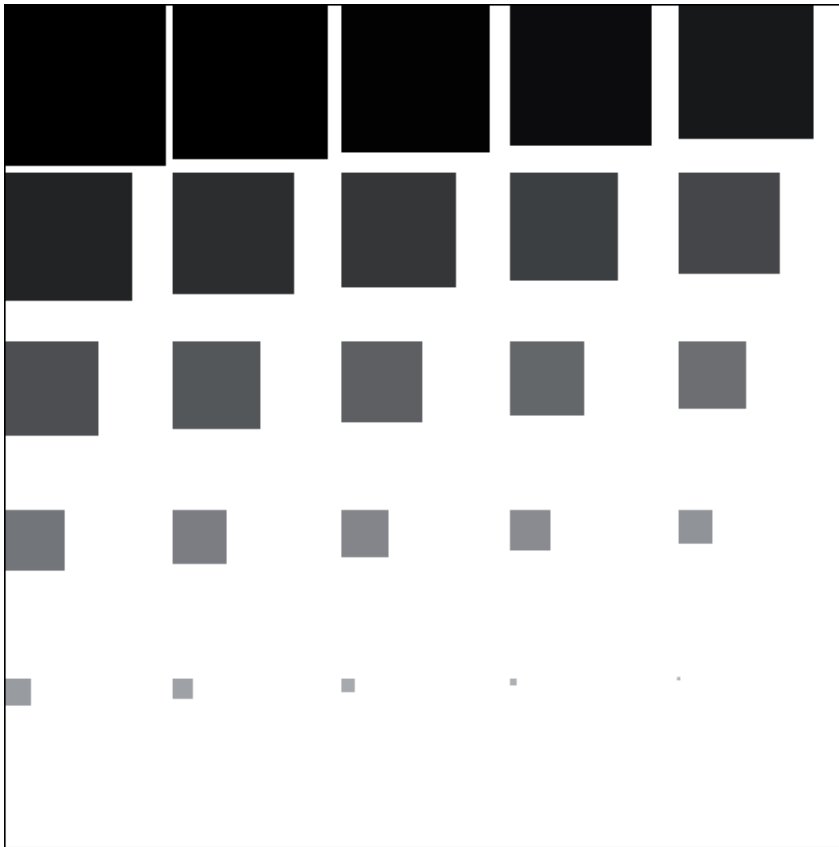
Dennis Blöte, 04.12.2006

### Grafische Entwicklung durch verschachtelte Iteration

```
void setup()
{
    // Vorgaben aus der Aufgabe
    int canvas_size = 500;
    int box_size = 100;
    // Variablen
    int start_size = 95;
    int size_factor = -4;
    int start_margin = 0;
    int margin_factor = 0;
    int start_color = 0;
    int color_factor = 8;
    // Setup
    size(canvas_size, canvas_size);
    background(255);
    // Funktionsaufruf
    draw_rects(canvas_size, box_size, start_size, size_factor,
               start_margin, margin_factor, start_color, color_factor);
}

void draw_rects(int cs, int bs, int ss, int sf,
               int sm, int mf, int sc, int cf)
{
    int act_size = ss;
    int act_margin = sm;
    int act_color = sc;
    // vertikale Schleife
    for(int i=0; i<(cs/bs); i++) {
        int y = i * bs + act_margin;
        // horizontale Schleife
        for(int j=0; j<(cs/bs); j++) {
            int x = j * bs + act_margin;
            fill(act_color);
            stroke(act_color);
            rect(x, y, act_size, act_size);
            act_size += sf;
            act_margin += mf;
            act_color += cf;
        }
    }
}
```

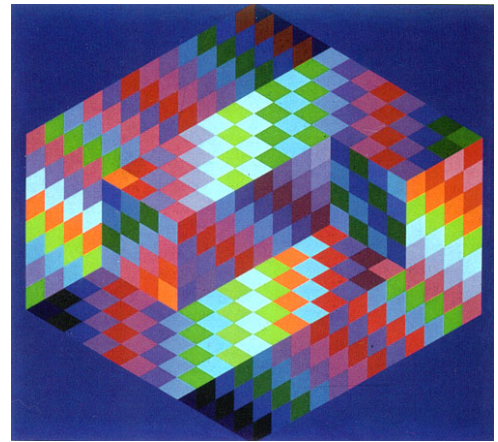
## Grafische Entwicklung durch verschachtelte Iteration



## Vasarely und das Raster

In Vasarelys Werken spielen Raster eine sehr große Rolle. Viele seiner Bilder haben als Hauptelemente Würfel und andere quadratische Gebilde. Diese sind oftmals so angeordnet, dass sich optische Täuschungen ergeben oder der Betrachter zumindest etwas länger benötigt, um sie ganz zu erfassen.

Die Anordnung der Würfel folgt bei dieser Darstellungsform stets einem isometrischen Raster, bei dem die Diagonalen im 45°-Winkel verlaufen. Dadurch erreichen Vasarely, dass die Bilder dreidimensional wirken und somit eine Tiefe bekommen.



„Lineaturen und Farbfelder werden in einer elementaren, beinahe mathematischen Gestaltung zusammengeführt und vom Künstler als Form-Module ('unités plastique') bezeichnet. Es entstehen schier undurchdringliche Raster von kleinen geometrischen Formationen [...]“ - Beschreibung zu Werken an der Ruhr Universität, Bochum [1]



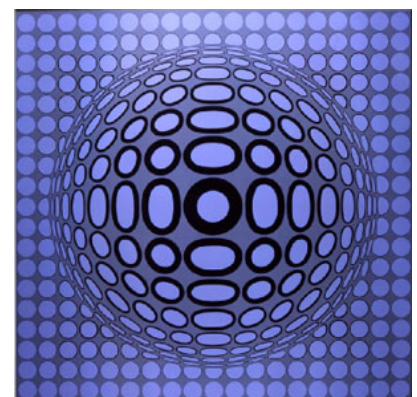
Andere Werke von Vasarely hingegen stellen diese Dreidimensionalität nicht so offensichtlich dar und wirken zunächst flach - diesen Bildern fehlt jeweils das isometrische Raster.

Nach längerer Betrachtung hingegen erhalten auch sie - wie im Beispiel links - eine Tiefe. Durch die immer kleiner werdenden Rechtecke, welche in diesem Fall auch gleichzeitig das Raster darstellen ergibt sich ein Art Tunnel - dieser Eindruck wird zusätzlich durch die Farbgebung der Rechtecke unterstützt.

Vasarelys Bilder enthalten jedoch nicht nur Quadrate, sondern auch Kreise: Dabei kommt es oftmals vor, dass sich diese wie Kugeln aus dem Bild heraus wölben.

Durch diese Verformung wird das eigentliche Raster des Bildes gebrochen - es lässt sich jedoch durch die in diesem Fall äußeren Punkte nachempfinden.

Verallgemeinernd könnte man sagen, dass Vasarelys Bilder auf sehr feinen Rastern aufbauen, welche nicht immer starr verfolgt, sondern teilweise auch bewusst gebrochen werden, um eine Dreidimensionalität entstehen zu lassen.



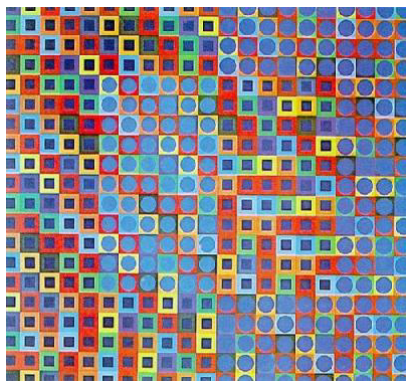
[1] [http://www.ruhr-uni-bochum.de/kuba/data/vasarely\\_1.htm](http://www.ruhr-uni-bochum.de/kuba/data/vasarely_1.htm)

## Wie ließen sich Vasarelys Bilder programmieren?

Da viele der Bilder auf einem Raster beruhen, wäre der Grundansatz, sie mit verschachtelten Iterationen zu programmieren, wie wir dies auch schon in der ersten Aufgabe gemacht haben. Dabei müssten jedoch je nach Komplexität des darzustellenden Bildes mehr oder weniger vielfältige Variationen implementiert werden.

Wenn wir von den eher einfacheren Bildern, wie dem Beispiel rechts ausgehen, so könnte man auch mit einer einfachen Iteration arbeiten, welche in diesem Fall immer kleiner werdende Rechtecke zur Mitte hin malt. Für die Farben der Rechtecke müsste man einen Ablauf vorgeben, anhand dessen sie sich entwickeln können.

Die statischste Methode in diesem Fall wäre es, die Farbwerte aufeinanderfolgend in einem Array zu hinterlegen. Man könnte die Farbwerte allerdings auch berechnen, da diese einem Muster zu folgen scheinen.



Bei Bildern wie beispielsweise dem auf der rechten Seite müsste man mit zwei verschachtelten Iterationen arbeiten, welche die im Raster angeordneten Würfel zeichnen.

Da sich in der Farb- und Formgebung nur teilweise eine klare Abfolge erkennen lässt (in den Diagonalen), könnte man hier versuchen, mit per zufall generierten Werten ein ähnliches Bild zu erzielen. Bei der Farbgebung der Rechteckrahmen müsste man die Farbwerte jedoch auf die Bereiche der hellen und satten Farben beschränken, wohingegen man für deren Inhalte nur Blautöne erlauben dürfte.

Eine weitere Gattung der Bilder wären jene, die zwar auf einem Raster aufbauen, sich aber eher schwer zu programmieren ließen. Für Bilder wie die unteren beiden müsste man meiner Meinung nach viele Variationen vorgeben und könnte nicht alles automatisch vom Computer berechnen lassen.

