

# Color By Numbers

**DEMO**



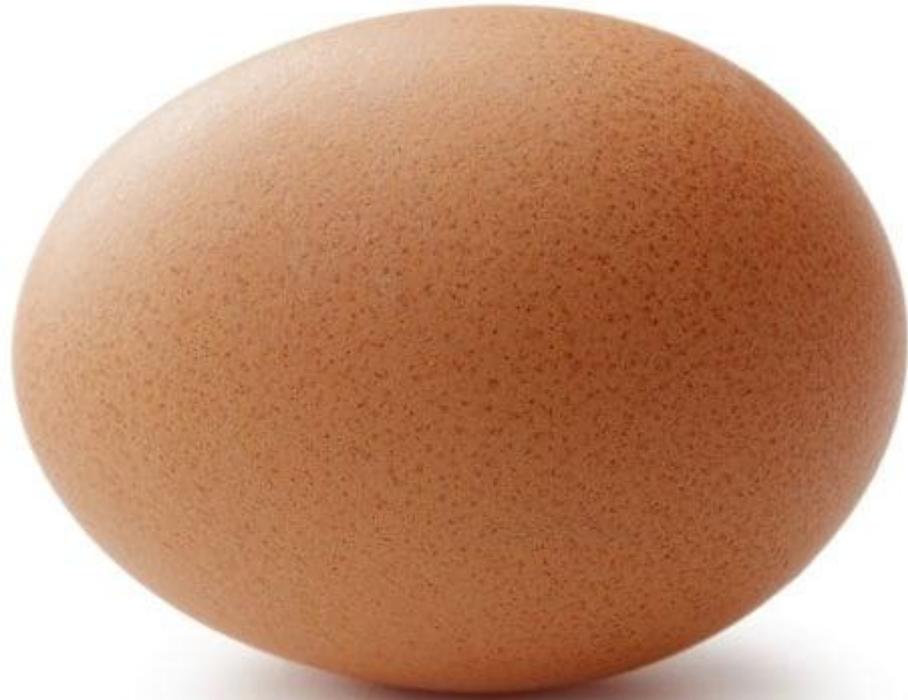
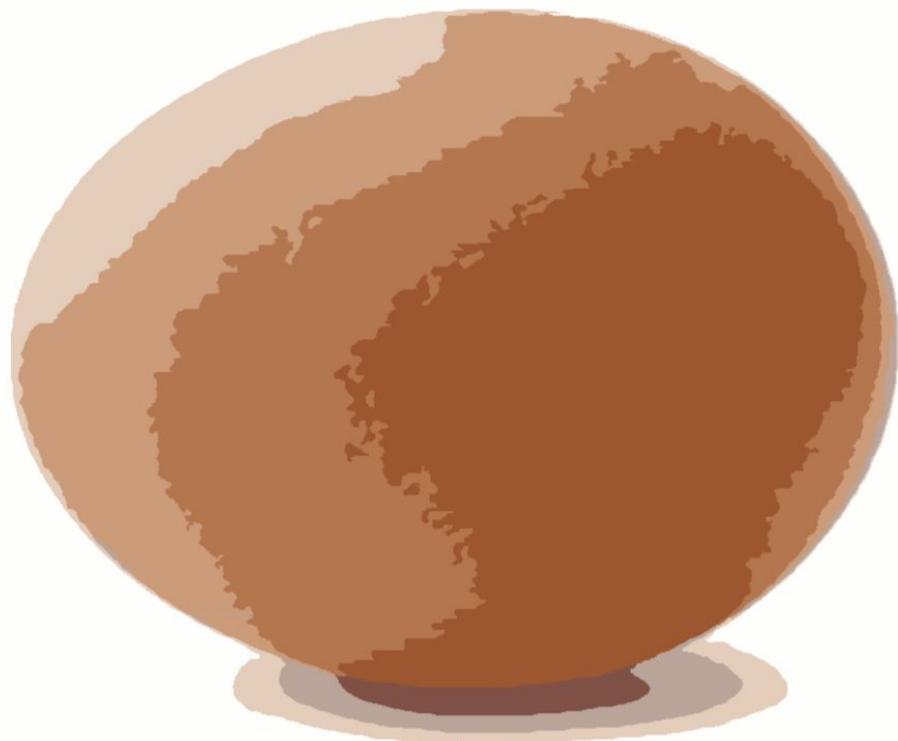
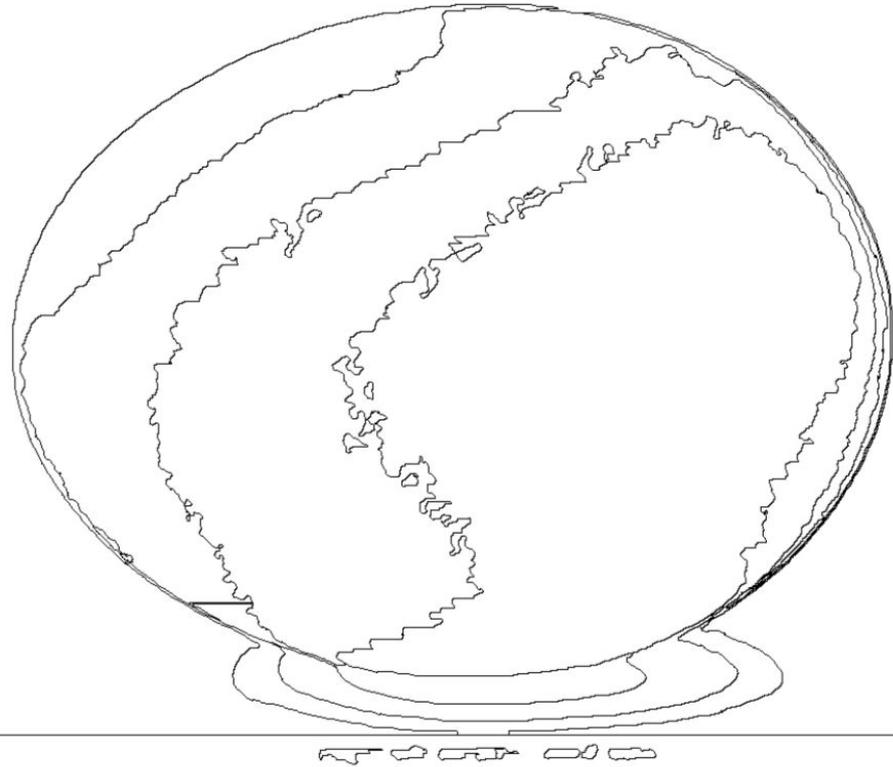
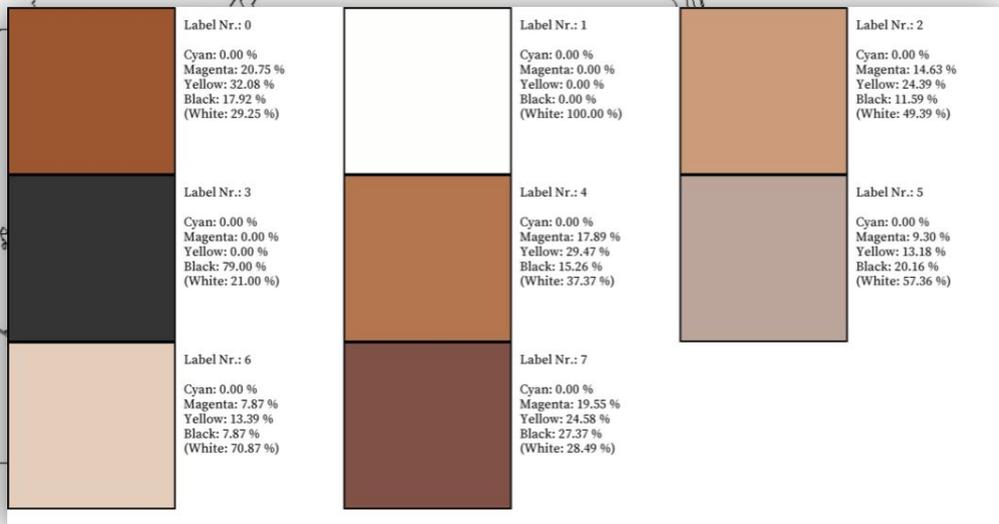


Image credit: [istockphoto.com/Floortje](https://www.istockphoto.com/Floortje)







# Farbreduzierung

---

# k-Means

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$$

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$$

# k-Means Clustering-Algorithmus

## Input

- $k \hat{=}$  Anzahl der Cluster (vorab festgelegt)
- $x_1 \dots x_n \hat{=}$  numerische Daten (Farbwerte)

# k-Means Clustering-Algorithmus

## Input

- $k \hat{=}$  Anzahl der Cluster (vorab festgelegt)
- $x_1 \dots x_n \hat{=}$  numerische Daten (Farbwerte)

Initialisierung der Cluster Werte (es existieren viele verschiedene Varianten)

- Zentroide  $c_1 \dots c_k$  werden zufällig gewählt (Farbwerte)
- k-Means++

# k-Means Clustering-Algorithmus

## Input

- $k \hat{=}$  Anzahl der Cluster (vorab festgelegt)
- $x_1 \dots x_n \hat{=}$  numerische Daten (Farbwerte)

Initialisierung der Cluster Werte (es existieren viele verschiedene Varianten)

- Zentroide  $c_1 \dots c_k$  werden zufällig gewählt (Farbwerte)
- k-Means++

## Kern des Algorithmus

1. finde für jeden Datenpunkt  $x_i$  den nächstgelegenen Zentroiden  $c_j$  (euklidischer Abstand)
2. jeder Datenpunkt  $x_i$  wird dem Cluster  $j$  zugeordnet (Zuweisung Zentroid  $c_j$ )
3. für jeden Cluster  $1 \dots k$  wird der Centroid neu berechnet, über den Durchschnitt aller dem Centroid zugeordneten Datenpunkte  $x$
4. Die Schritte 1. - 3. werden so lange wiederholt, bis sich die Centroide nicht mehr ändern.

# k-Means Clustering-Algorithmus

## Input

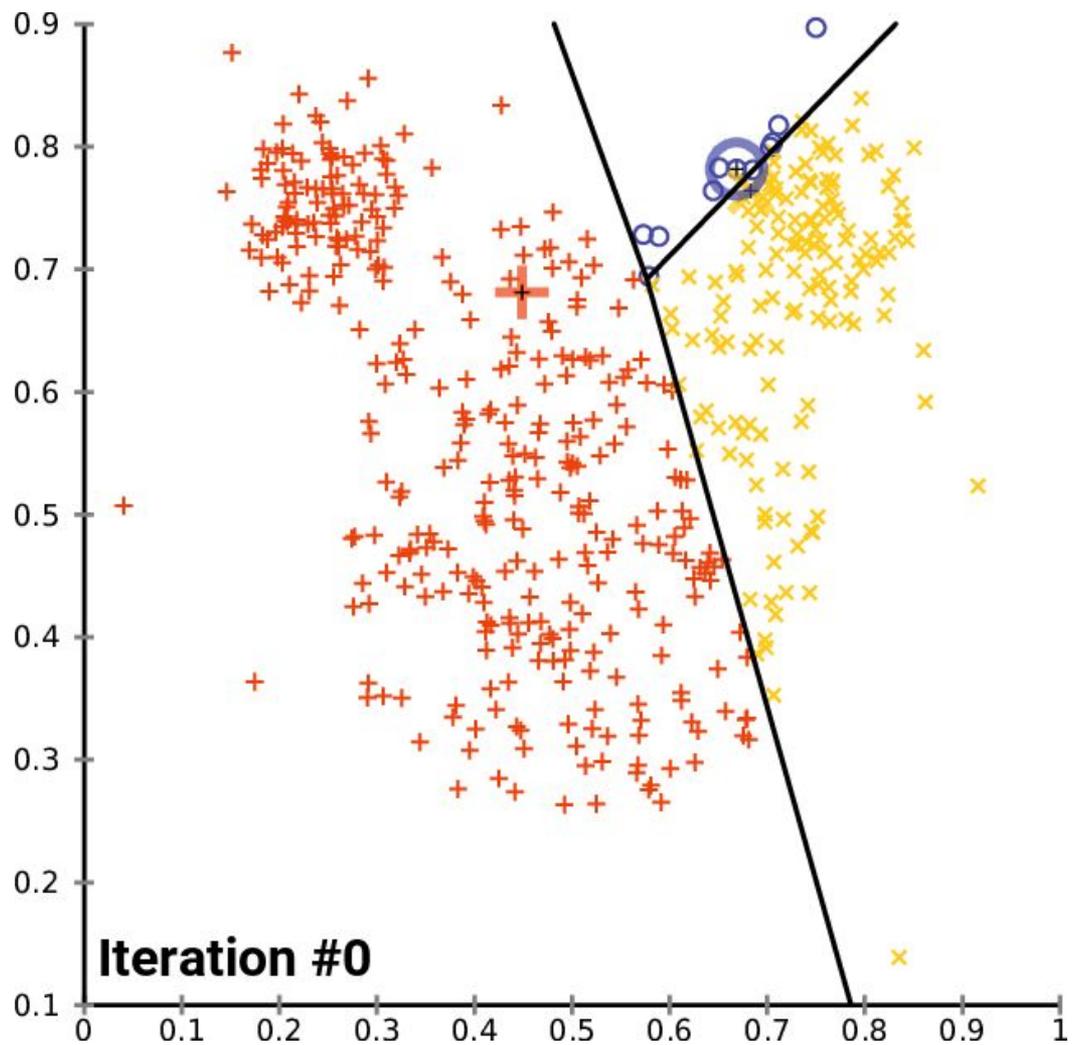
- $k \hat{=}$  Anzahl der Cluster (vorab festgelegt)
- $x_1 \dots x_n \hat{=}$  numerische Daten (Farbwerte)

Initialisierung der Cluster Werte (es existieren viele verschiedene Varianten)

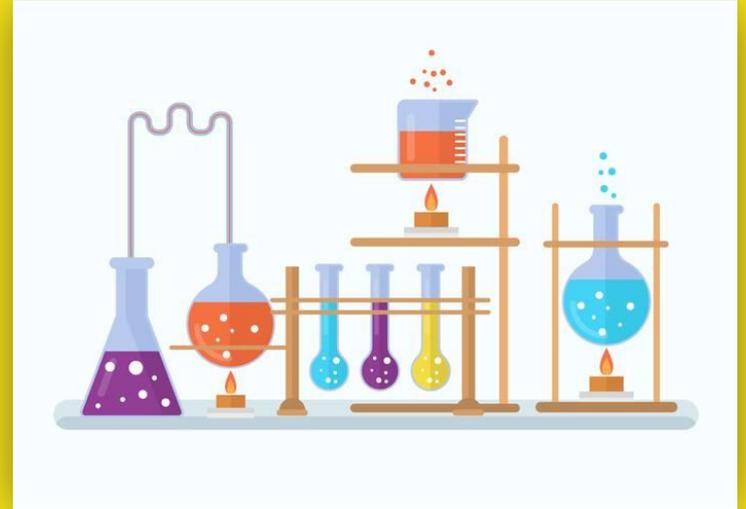
- Zentroide  $c_1 \dots c_k$  werden zufällig gewählt (Farbwerte)
- k-Means++

## Kern des Algorithmus

1. finde für jeden Datenpunkt  $x_i$  den nächstgelegenen Zentroiden  $c_j$  (euklidischer Abstand)
2. jeder Datenpunkt  $x_i$  wird dem Cluster  $j$  zugeordnet (Zuweisung Zentroid  $c_j$ )
3. für jeden Cluster  $1 \dots k$  wird der Centroid neu berechnet, über den Durchschnitt aller dem Centroid zugeordneten Datenpunkte  $x$
4. Die Schritte 1. - 3. werden so lange wiederholt, bis sich die Centroide nicht mehr ändern.



# Farbreduzierung Tests



—

**Original**



**Neu k=16**



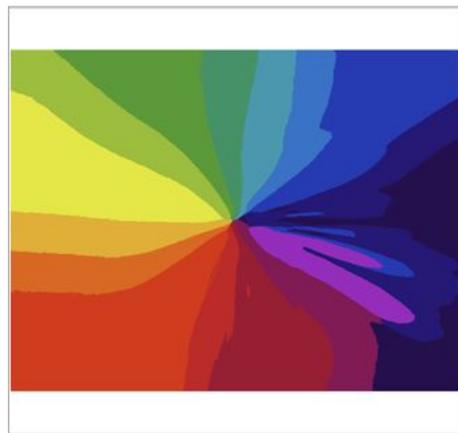
**k=8**



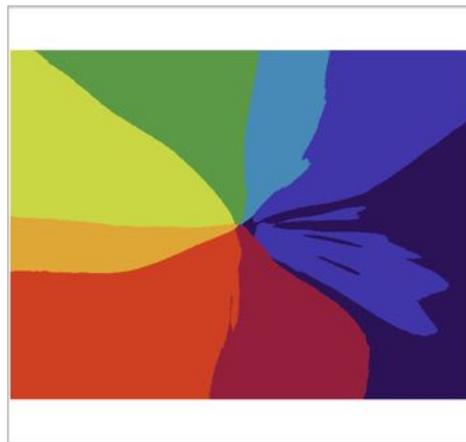
Original



Neu k=16



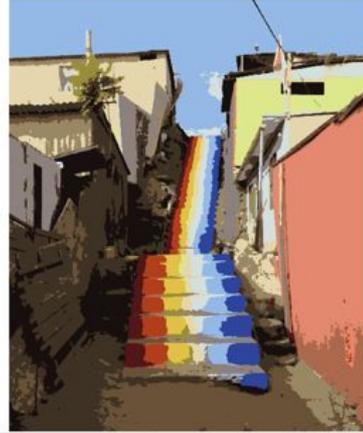
k=8



Original



$k=16$



**Original**



**k=16**



**k=8**



Original



k=16



k=8



Original



k=16



k=8



**Original**



**k=16**



**k=8**



Original



k=16



k=8



**Original**



**k=16**



**k=8**



**Original**



**k=16**



**k=8**



Original



k=16



k=8



**Original**



**k=16**



**k=8**



Original



k=16



k=8



**Original**



**k=16**



**k=8**



**Original**



**k=16**



**k=8**



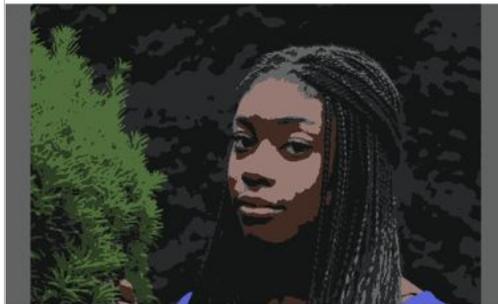
**Original**



**k=16**



**k=8**



**Original**



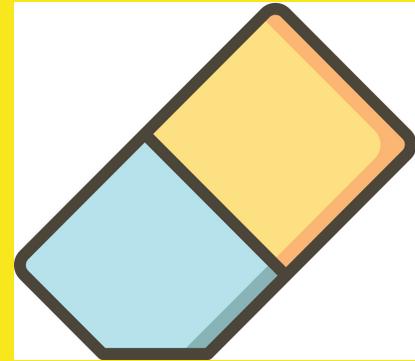
**k=16**



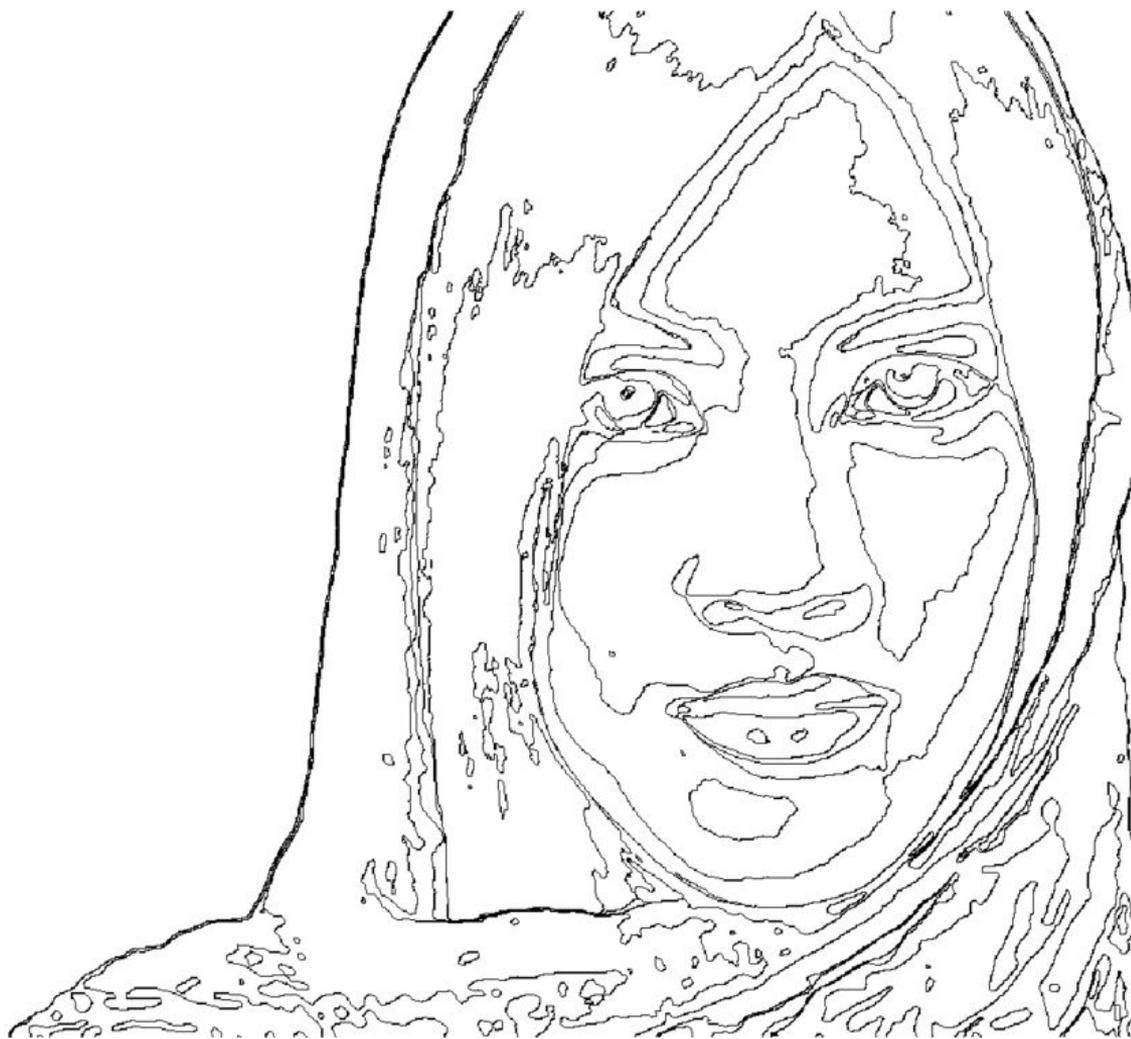
**k=8**

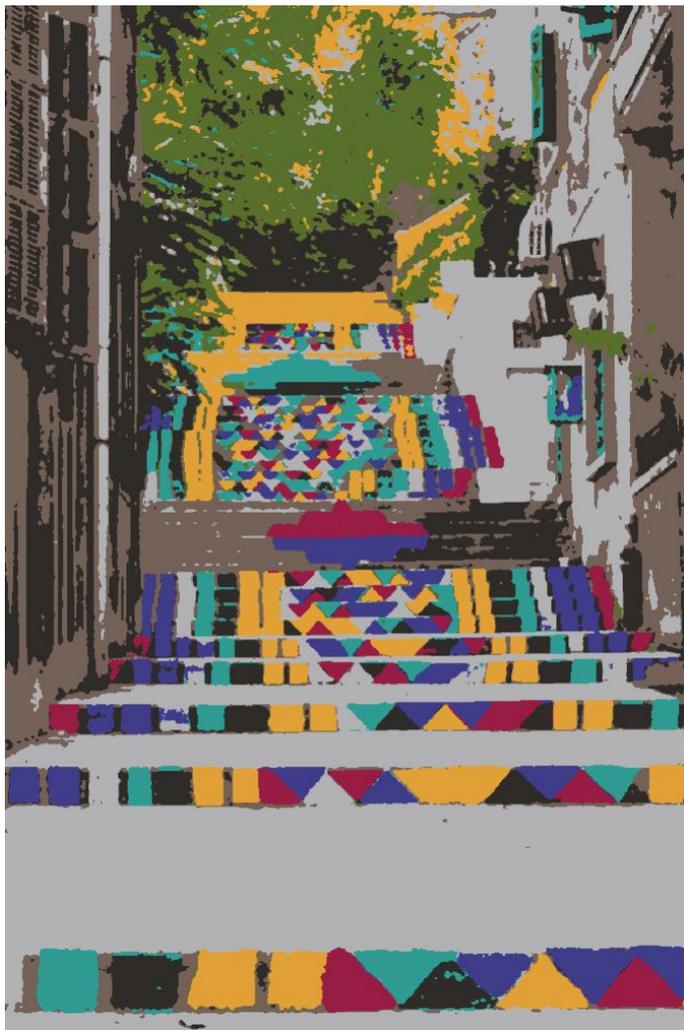


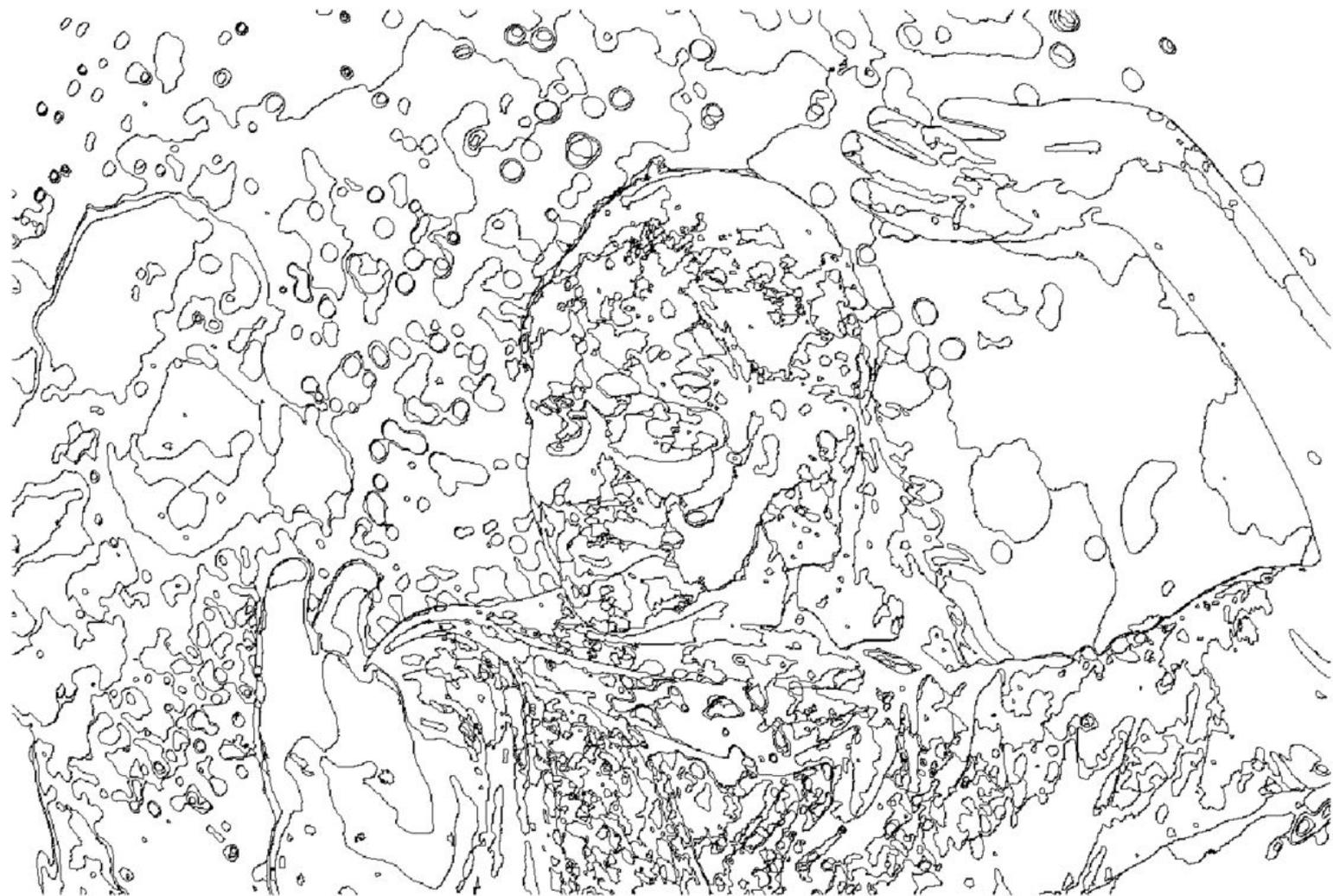
# Entfernung zu kleiner Flächen



—

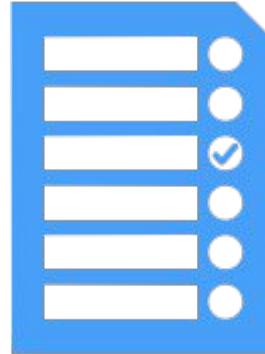






# Flächenbestimmung

Mit der Farbreduktion wird eine Liste von Labeln erzeugt, die für jeden Pixel im Bild angibt, welche Farbe er hat



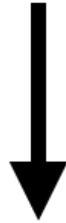
# Flächenbestimmung

Label pro Zeile zusammenfassen



# Flächenbestimmung

Zeilen verschmelzen



# **Zu kleine Flächen verschmelzen**

**Solange nicht alle Flächen  
mindestens die minimale Fläche aufweisen,  
verschmelzen die zu kleine Flächen miteinander**

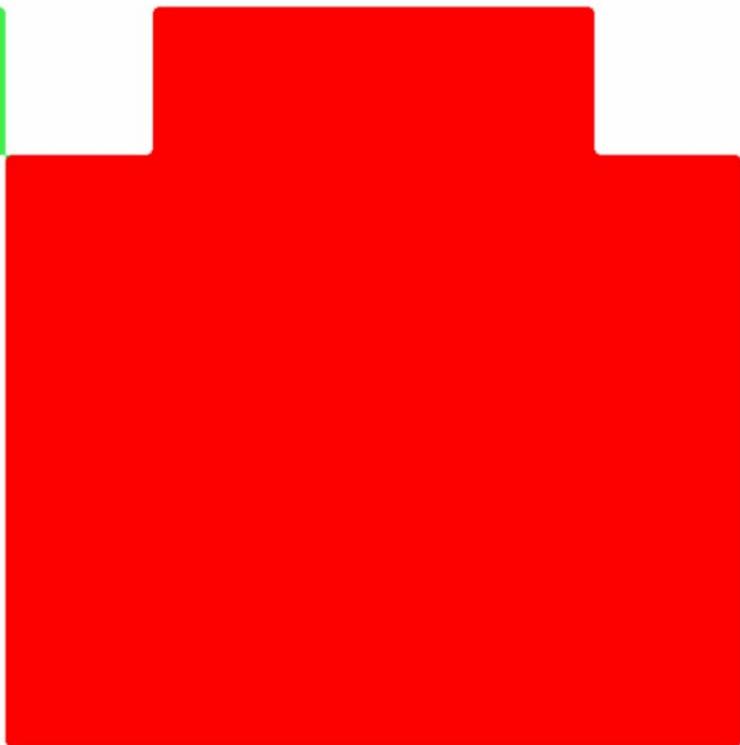
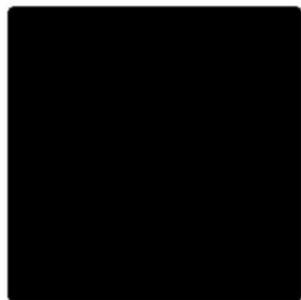
**Zahl setzen**

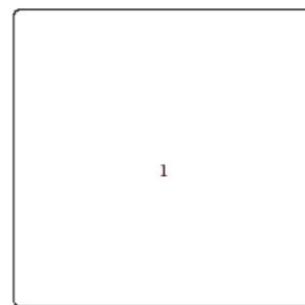
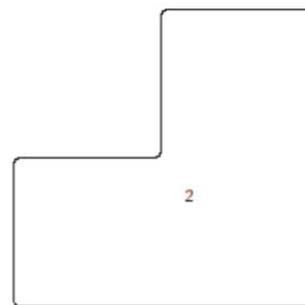
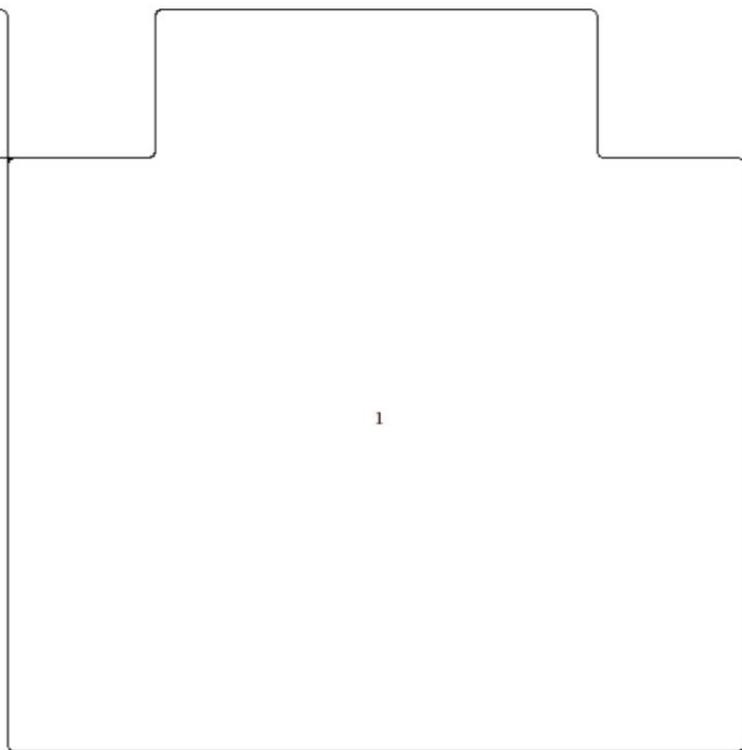
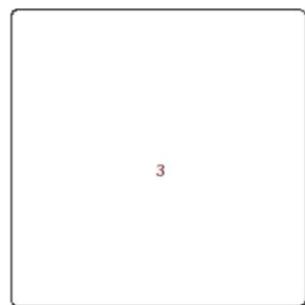
**1**

**2**

**3**

—





**Danke für die  
Aufmerksamkeit**

---