

## Wdh. Bild aus Datei laden

*PImage img;* (*PImage* ist eine Klasse mit verschiedenen Methoden)

```
void setup() {
    img = loadImage(„bild.jpg“);
    size(img.width, img.height);
}

void draw() {
    //Image(img, 0, 0);
    loadPixels();

    img.loadPixels();

    for(int x=0; x<width, x++) {
        for(int y=0, y<height, y++) {
            int index = y*width+x;
            color pixel = img.pixels[index];
            //Bildbearbeitung einfügen
            pixels[index] = pixel;
        }
    }
    updatePixels();
}
```

PImage	Klasse
width height	Variablen der Klasse
resize() loadPixels() updatePixels()	Funktionen der Klasse (Methode)

Beispiel:

```
img = loadImage(...);
img.resize(200,300);
```

## Laden der Pixel eines Bildes in das Array pixels[]

Normalerweise wird ein auf dem Bildschirm vorhandenes Bild mit `loadPixels()` und `updatePixels()` verändert. Es wird dabei bei `loadPixels()` jenes Array von Pixeln geladen, welches die aktuell auf dem Bildschirm angezeigten Pixel enthält.

Eine andere Variante ist das Laden des Arrays `img.pixels[]` aus einem geladenen Bild. Dessen Inhalt kann nun unabhängig vom Array `pixels[]` verändert werden und auch in dieses Array kopiert werden.

## Effekte der Bildbearbeitung

### *Änderung der Helligkeit*

Addition eines konstanten Werts zu allen Pixelwerten.

### *Änderung des Kontrasts*

Multiplikation aller Pixel mit einem konstanten Wert

### *Umwandlung in Graustufenbild*

Für jedes Pixel: Bildung des Mittelwerts aus Rot, Grün und Blau.

### *Umwandlung in Binärbild*

1. Umwandlung in Graustufenbild
2. Schwellwertbildung (alle Pixel über der Schwelle auf weiß, alle anderen Schwarz)

## Beispiele / Wiederholung zur Klausurvorbereitung

Ablauf eines zeitgesteuerten Programms:

*//Deklaration der Variablen*

*int pos;*

*//Initialisierung*

```
void setup() {  
    pos = 0;  
    size (200,200);  
}
```

```
void draw() { // wird alle 1/60 Sekunde ausgeführt  
    rect(pos, pos, 10,10);  
    pos++;  
}
```

Zeichenfunktionen:

*rect(), ellipse(), line()*

*fill(), stroke()*

Variablen müssen vor der Verwendung mit Variablenname und Variablentyp deklariert werden.

Kontrollstrukturen:

```
if(gray > schwelle) {  
...  
}  
else {  
...  
}
```

```
for(int i=0; i<width; i++){  
...  
}
```

