

In der ImageJ-Klasse `ij.plugin.filter.GaussianBlur` ist ein einfaches Gauß-Filter implementiert, dessen Radius (σ) frei spezifiziert werden kann. Dieses Gauß-Filter ist natürlich mit separierten Filterkernen implementiert (s. Abschn. 6.3.3).⁴ Hier ist ein Beispiel für dessen Anwendung:

```

1  import ij.plugin.filter.GaussianBlur;
2  ...
3  public void run(ImageProcessor I) {
4      GaussianBlur gb = new GaussianBlur();
5      double radius = 2.5;
6      gb.blur(I, radius);
7  }
```

6.6.3 Nichtlineare Filter

Ein kleiner Baukasten von nichtlinearen Filtern ist in der ImageJ-Klasse `ij.plugin.filter.RankFilters` implementiert, insbesondere Minimum-, Maximum- und gewöhnliches Medianfilter. Die Filterregion ist jeweils (annähernd) kreisförmig mit frei wählbarem Radius. Hier ein entsprechendes Anwendungsbeispiel:

```

1  import ij.plugin.filter.RankFilters;
2  ...
3  public void run(ImageProcessor I) {
4      RankFilters rf = new RankFilters();
5      double radius = 3.5;
6      rf.rank(I, radius, RankFilters.MIN); // Minimum Filter
7      rf.rank(I, radius, RankFilters.MAX); // Maximum Filter
8      rf.rank(I, radius, RankFilters.MEDIAN); // Median Filter
9  }
```

6.7 Aufgaben

Aufg. 6.1. Erklären Sie, warum das „Custom Filter“ in *Adobe Photoshop* (Abb. 6.6) streng genommen kein lineares Filter ist.

Aufg. 6.2. Berechnen Sie den maximalen und minimalen Ergebniswert eines linearen Filters mit nachfolgender Filtermatrix $H(i, j)$ bei Anwendung auf ein 8-Bit-Grauwertbild (mit Pixelwerten im Bereich $[0, 255]$). Gehen Sie zunächst davon aus, dass dabei kein *Clamping* der Resultate erfolgt.

⁴ Zur Implementierung in ImageJ ist anzumerken, dass die in der Methode `blur()` generierten Filterkerne relativ zum angegebenen Radius zu klein dimensioniert werden, und es dadurch zu erheblichen Fehlern kommt.