

Bildbearbeitung

Team:

Dominik Grasser k01456254

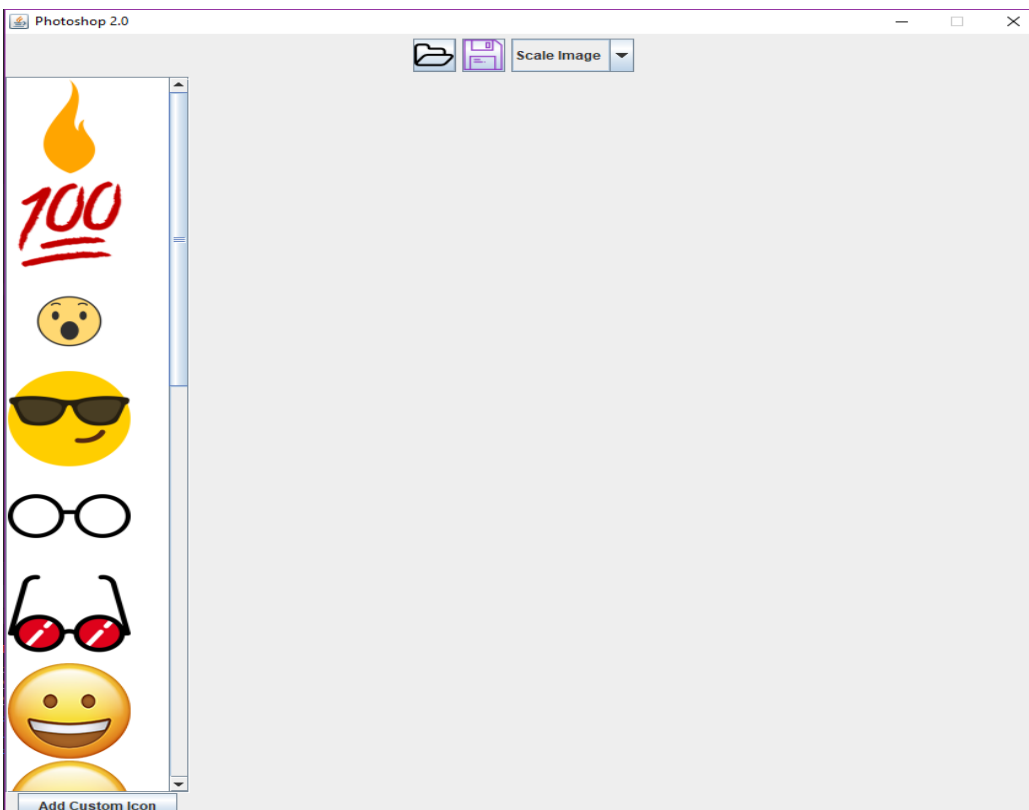
Carl Weilguny k1607833

Philipp Kern k01555242

Beschreibung:

Unser Programm soll dem Benutzer ermöglichen seine Lieblingsbilder noch weiter zu verbessern. Mithilfe unseres Programms kann man bestimmte Dinge zu seinem Bild hinzufügen. Der Benutzer wählt eines von seinen Bildern aus , mithilfe unseres Programms kann er dann bestimmte Filter darauf anwenden und kleine Symbole , Zeichen oder auch Accessoires hinzufügen. Anschließend kann er noch die Größe ändern. Ist der Benutzer fertig wird das Bild gespeichert und ist bereit an Freunde weitergeschickt zu werden.

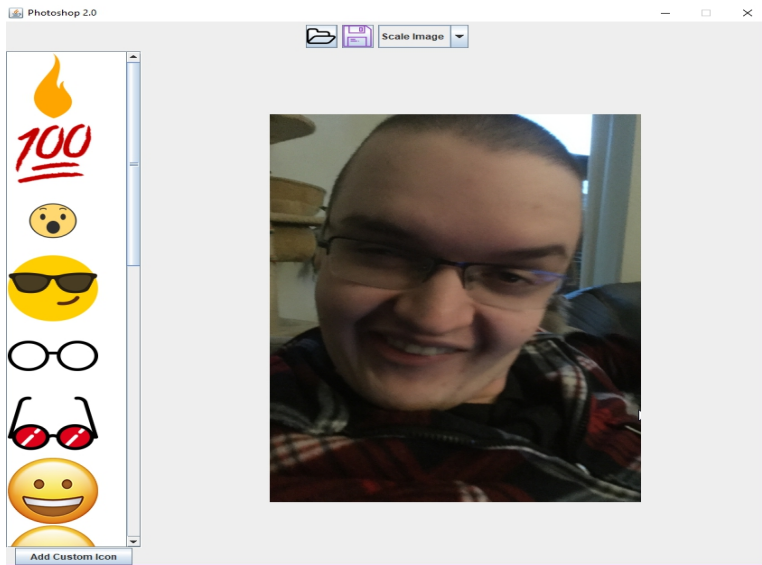
Überblick über die Entwicklung:



Main app:

Wir haben die Main-App sehr schlicht, aber dafür übersichtlich gehalten, damit der Benutzer das Programm versteht und schnell das Bild bearbeiten kann. Im Fenster sind drei Knöpfe , eine Box gefüllt mit verschiedensten „Emojis“ und eine Dropdown-Liste zu sehen. Um die Bearbeitung des Bildes zu starten, muss der Knopf mit dem Datei-Symbol geklickt werden, der sich oben in der Mitte befindet. Wird dieser Knopf gedrückt, öffnet sich ein Fenster mit dem ein .jpg oder .png Bild ausgewählt werden kann. Hat der Benutzer das gewünschte Bild gefunden, wird es geöffnet und im Fenster angezeigt. Der nächste Knopf mit dem Speicher-Symbol, speichert das geöffnete Bild, wenn er gedrückt wird. Der Speicherort und der Name kann beliebig gewählt werden, doch wichtig ist, dass beim Dateinamen „.jpg“ oder „.png“ angehängt wird.

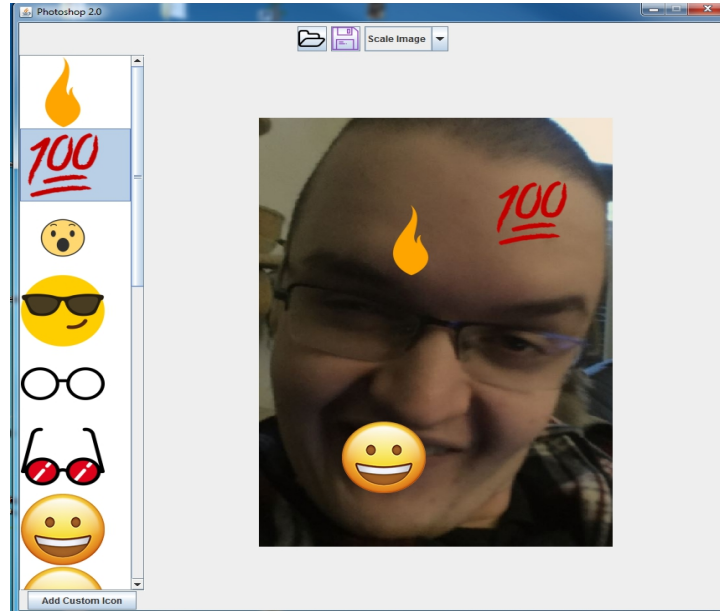
Screenshot sobald der Benutzer ein Bild ausgewählt hat.



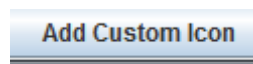
Emojis:

Auf der linken Seite befindet sich eine Liste von verschiedensten Emojis die von uns schon bereitgestellt sind.

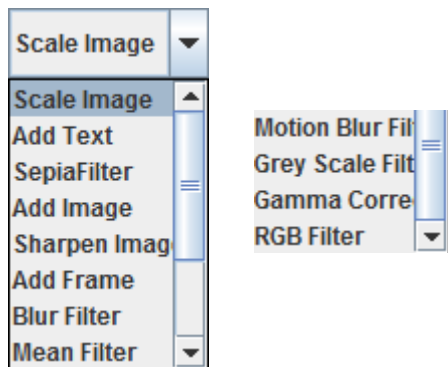
Die Emojis können dann mittels Drag and Drop ganz bequem in das Bild hineingezogen werden.



Darunter gibt es auch noch einen Knopf mit dem man ein selbst gewähltes Bild hinzufügen kann.



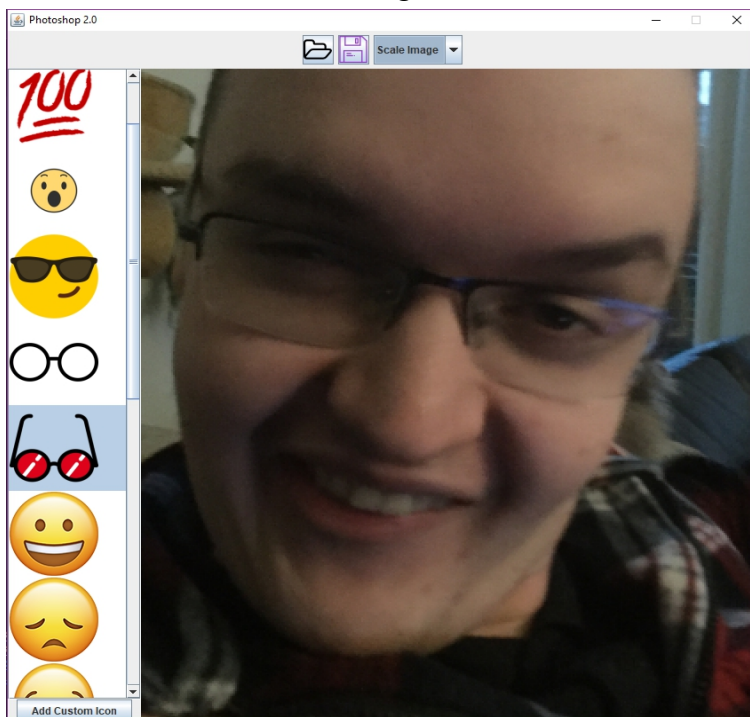
Will der Benutzer nun das Bild bearbeiten, so können verschiedenste Funktionen in der Dropdown-Liste ausgewählt werden.



Scale Image:

Mit dieser Funktion kann das gewählte Bild vergrößert werden. Der Benutzer wird nach einem Scaling Factor abgefragt, bedeutet wenn als Factor zwei gewählt wird, verdoppelt sich die Größe des Bilds.

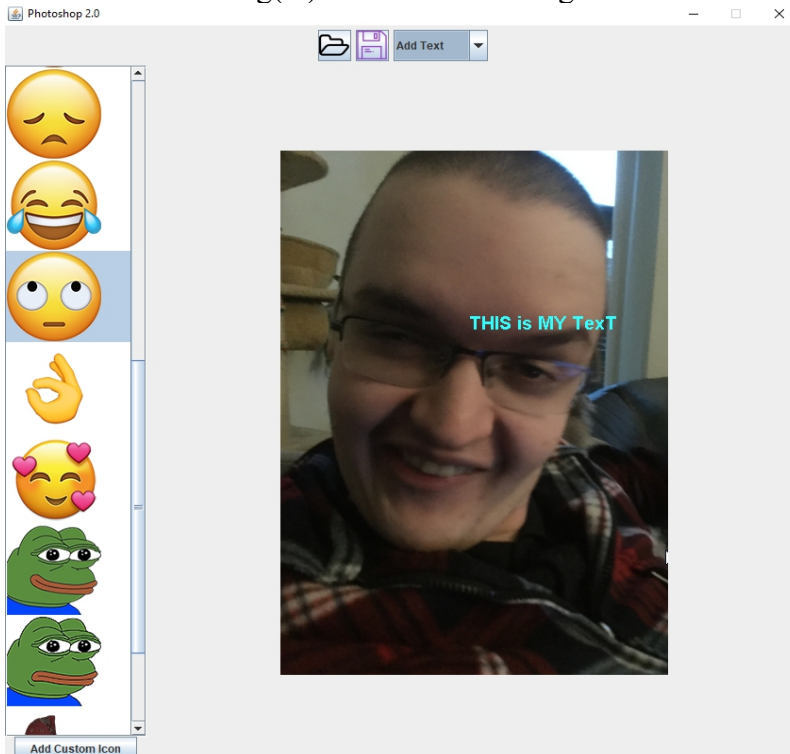
Die folgende Methode wurde wie folgt implementiert: Der Wert der vom Benutzer eingegeben wurde, wird als String eingelesen und zu einem Integer umgewandelt. In der ImageHelper Klasse befindet sich eine sogenannte `scaleImage(BufferedImage image, int maxWidth, int maxHeight, int quality)` Methode, die uns hilft das gewählte Bild zu skalieren. Die Parameter für die Methode werden so gewählt, dass `maxWidth` und `maxHeight` die Breite und Höhe mal dem Factor beträgt. Die Qualität ist `Image.SCALE_SMOOTH`. Nachdem das Bild durch die Methode skaliert wurde, wird das alte Bild vom Frame removed, das neue hinzugefügt und neu gezeichnet. Returned wird das neue skalierte `BufferedImage`.



Add Text Command:

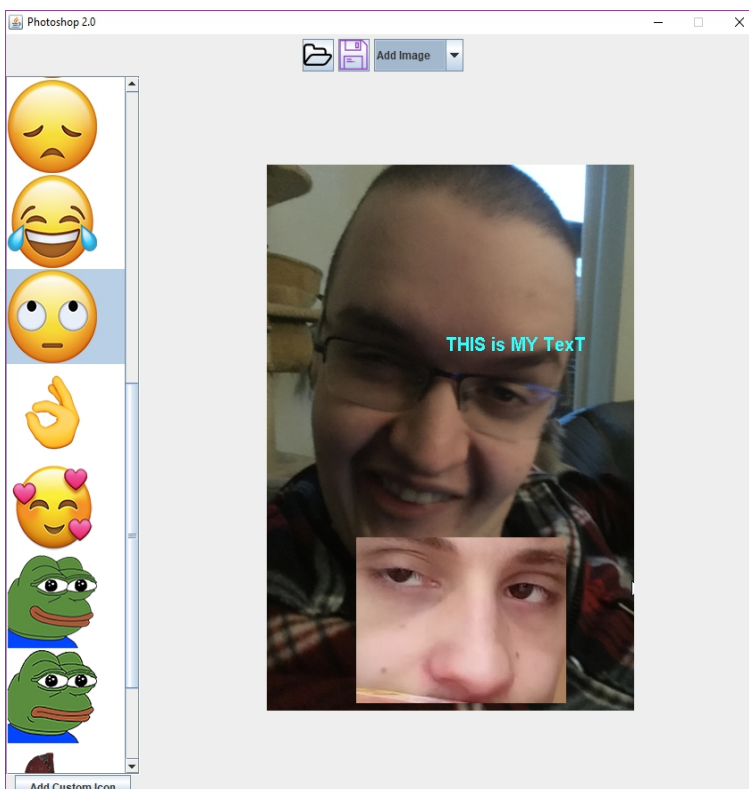
Eine weitere Funktion unseres Programms ist, einen beliebigen Text hinzuzufügen. Der Benutzer muss seinen gewünschten Text eingeben, die Position der X und Y Koordinate, die gewünschte Größe des Texts und zu guter letzt, die Farbe. Wenn alles korrekt eingegeben wurde, wird der Text

im Bild an der richtigen Position angezeigt. Um diese Funktion zu realisieren, benötigt man die Graphics des Bildes, die man sich bei jedem BufferedImage einfach mit `getGraphics()` holen kann. Zuerst wird das Bild wieder gezeichnet, dann die Farbe und die Schriftart des Texts gewählt und durch die `drawString(...)` Methode der String and der Position die gewählt wurde, gezeichnet.



Add Image Command:

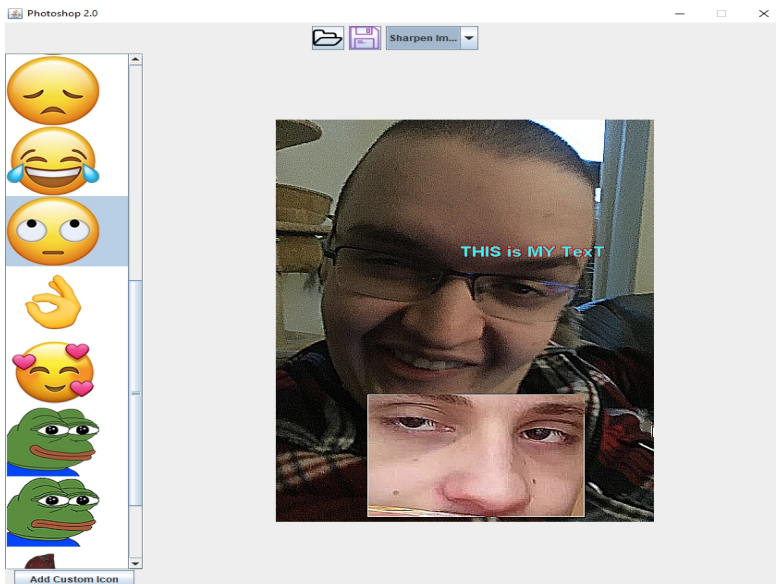
Das hinzufügen eines Bildes ist auch kein Problem. Im Grunde ist das hinzufügen eines Bildes sehr ähnlich dem hinzufügen eines Textes. Man wählt ein Bild von seinem Computer aus, gibt die Position an und schon ist das Bild durch ein weiteres Bild erweitert. Die Implementierung dieser Funktion und der Add text Funktion ist fast gleich. Nur wird hier kein String gezeichnet, sondern wieder mit der `drawImage(...)` Methode ein weiteres Bild über dem Grundbild gezeichnet.



Sharpen Image:

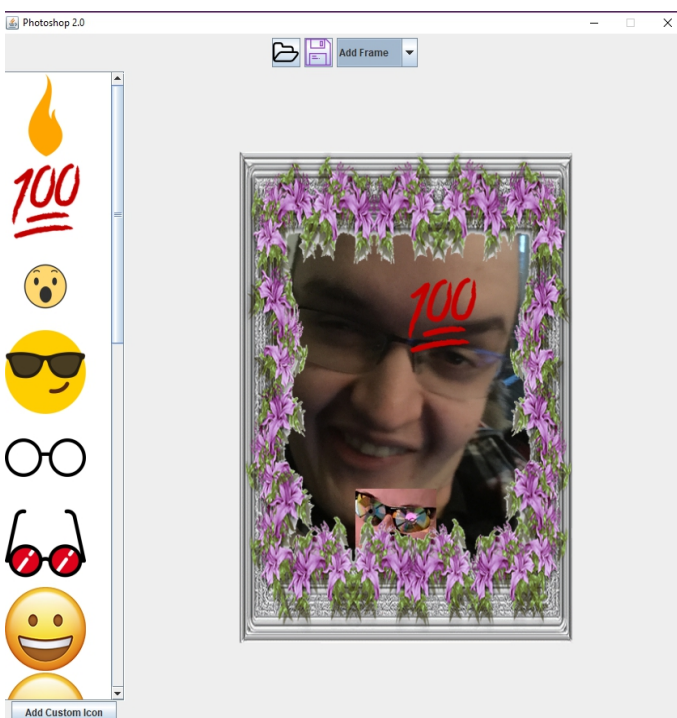
Falls der Benutzer das Bild schärfer haben möchte, so kann er diese Funktion benutzen. Ein kurzer Klick auf das Sharpen Image Command und das Bild wird schärfer.

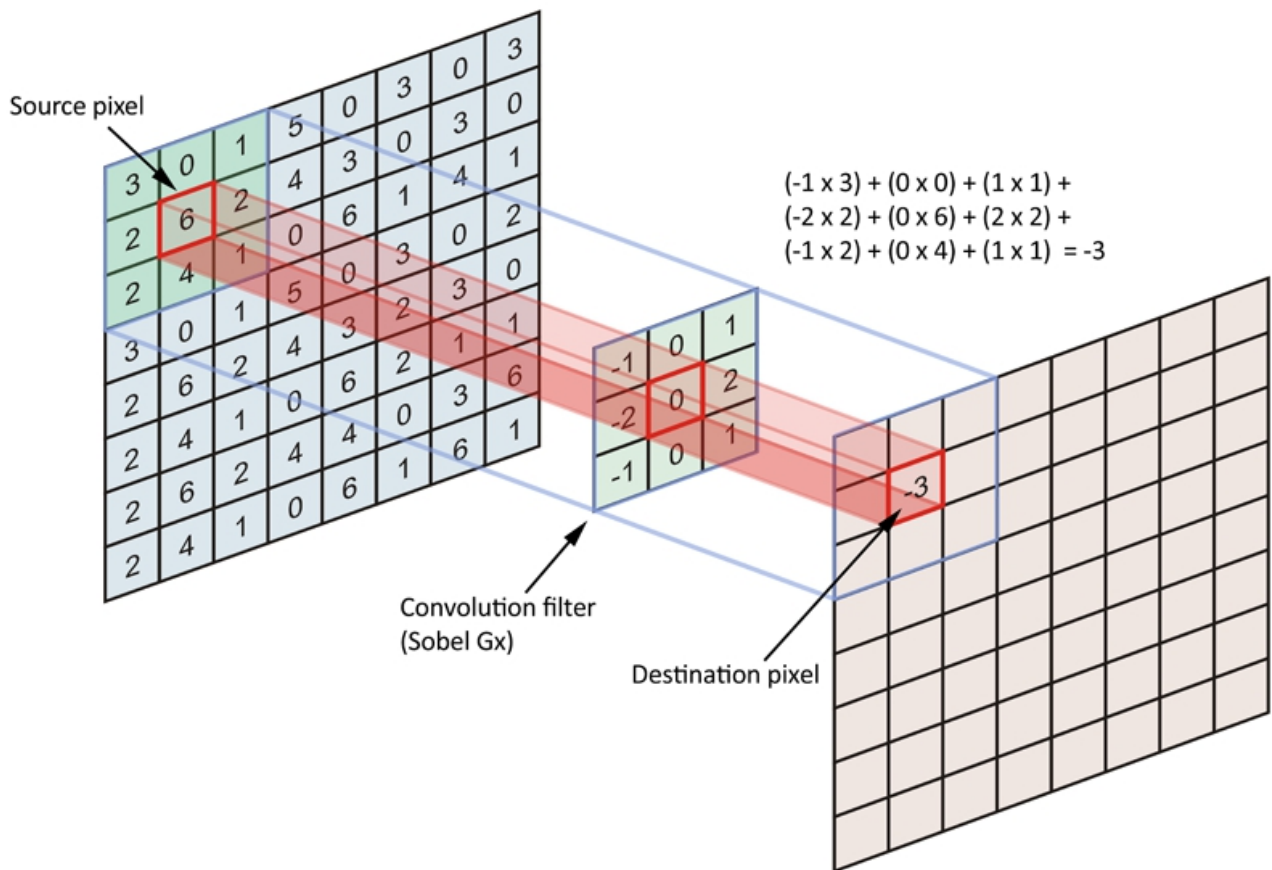
Dafür wird eine 3x3 Matrix angelegt die mit den Werten befüllt wird, sodass das Bild schärfer wird. Die Graphics des Bildes wird mit `getGraphics` geholt und das Bild wird mit `drawImage` neu gezeichnet. Nun erstellt man ein `BufferedImageOp` mit der gewünschten Matrix die wir schon erstellt haben und filtert das gezeichnete Bild mit der `filter(..)` Methode.



Add Frame Command:

Das fertige bearbeitete Bild kann auch noch am Schluss schön eingerahmt werden. Mit dem Add Frame Command kann sich der Benutzer einen Rahmen aussuchen, und dieser wird automatisch der Größe des Bildes angepasst und hinzugefügt. Das automatische anpassen der Größe wird durch die `getScaledInstance(...)` Methode erreicht, die als Parameter die Höhe und Breite des GrundBildes erhält. Dann holt man sich die Graphics des Frames und die Graphics des Bildes, dass eingerahmt werden soll, und zeichnet zuerst den skalierten Rahmen neu und danach wird der Rahmen auf das Grundbild gezeichnet.





Filter

Convolution Filter

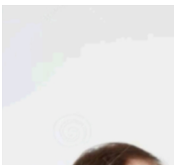
Der Convolution Filter funktioniert indem man eine Convolution Matrix (normalerweise $N \times N$) auf jedes Pixel eines Bildes anwendet.

Man zentriert die Matrix auf das aktuell durchzulaufende Pixel und multipliziert die umliegenden Pixelwerte mit den Werten der Matrix und addiert diese zusammen. Das Ergebnis wird dann in das durchzulaufende Pixel eingetragen.

Mithilfe dieser Technik lassen sich einige Effekte realisieren welche gleich demonstriert werden, doch zuvor noch das Originalbild:

Mean Filter:

Ist gedacht um das Bild zu glätten. Indem der Mittelwert der umliegenden Pixel berechnet wird.



Blur Filter:

Ist gedacht um das Bild zu verschwimmen.



Motion Blur Filter:

Ist gedacht um das Bild so aussehen zu lassen als ob das Objekt gerade in Bewegung ist.



Gamma Correction Filter:

Wird verwendet um die Helligkeit eines Bildes anzupassen. Desto kleiner gamma, desto heller das Bild, desto größer gamma, desto dunkler das Bild.

Dies wird unter anderem über die Bildserie in folgendem Artikel anschaulich:

https://en.wikipedia.org/wiki/Gamma_correction

Grey Scale Filter

Erzeugt von einem farblichen Bild ein Graustufen-Bild, indem es den Mittelwert der RGB-Werte berechnet



RGB Filter:

Mit diesem Filter kann man die RGB Werte prozentuell anpassen und somit eine Farbkorrektur erhalten. So können dann beispielsweise solche Bilder entstehen:



Sepia Filter:

Dieser Filter bildet einen Effekt nach, welcher auf alten analog Fotografien zu beobachten ist. Hierzu muss man die RGB Werte der einzelnen Pixel mit einem gewissen Faktor Gewichten.



Quellen:

Multimediasysteme Vorlesungsfolien Thema Bild

Multimediasysteme Übungsfolien 02: Bilder, Filter, Subsampling, Chromasubsampling

<https://www.dyclassroom.com/image-processing-project/how-to-get-and-set-pixel-value-in-java>

<https://docs.oracle.com/javase/tutorial/2d/images/>