

BIV-Übung 3

Einführung zu MATLAB

MATLAB erlaubt eine interaktive Befehlseingabe, was besonders ein schnelles Ausprobieren erleichtert. Hilfe zu einzelnen Funktionen können Sie durch den Befehl `doc <Funktionsname>` abrufen oder einfach durch Positionieren des Cursors auf dem Befehl und drücken von **F1**.

Im Editor können Sie mit doppelten Kommentarzeilen (`%%`) *Blöcke* („sections“) definieren, die einzeln abgearbeitet werden können (Start über grün/gelbe Icons „Run section“ oder „Run and Advance“). Diese Blöcke werden auch von der „Publish“ Funktion von Matlab zur Einteilung der Ausgaben in Abschnitte verwendet. Markierte *Zeilen* können durch Drücken von **F9** ausgeführt werden.

Auch für dieses Aufgabenblatt wird *keine Schleife* benötigt!

Protokoll In den Aufgabenstellung unten werden Sie aufgefordert Ihre Ergebnisse zu protokollieren. Sie können das mit der „Publish“ Funktion von Matlab machen oder legen Sie mit einem Textverarbeitungssystem Ihrer Wahl ein einfaches Dokument an und kopieren Sie die verwendeten Befehle und Screenshots der Ergebnisse in dieses Dokument.

1 Histogramme und globaler Kontrast

Auf der Moodle-Seite finden Sie eine zip-Datei mit einem Bild zu dieser Übung. Damit Sie Funktionen verwenden können, müssen sich die Dateien in Ihrem Arbeitsverzeichnis befinden oder Sie müssen das Verzeichnis mit den Dateien in den Suchpfad von MATLAB aufnehmen (Befehl `path` oder Menu File/SetPath).

- a) Laden Sie das Bild `blueten.jpg` mit `imread`. Um später keine Unterschiede zwischen Bildern mit Datentyp `uint8` (Wertebereich [0,255]) (wie das gerade geladene Bild) und `double` Bildern machen zu müssen, ist es sinnvoll, das Bild gleich in `double` zu konvertieren:

`doubleBild=double(uint8Bild)`. Den Wertebereich jetzt noch auf [0,1] anpassen.

Lassen Sie sich das Bild noch *nicht* anzeigen!

- b) Berechnen Sie die Histogramme der Farbkanäle R, G, B und lassen Sie sie alle gleichzeitig in Form von überlagerten Kurven anzeigen. Anleitung:

- Zunächst ein Fenster mit dem Befehl `figure` öffnen.
- Histogramme von Bildern mit *einem* Kanal (!) können mit `imhist` berechnet werden. Die Funktion gibt ein Array mit den Histogramm-Werten zurück. Wenn Sie den Rückgabewert *nicht* verwenden, wird das Histogramm direkt angezeigt. Für die nächste Teilaufgabe brauchen Sie jedoch das zurückgegebene Array!
- Kurven (bestehend aus miteinander verbundenen Punkten eines Arrays) lassen sich mit `plot(array, 'farbe')` anzeigen; zum Beispiel für rot 'farbe' auf 'r' setzen.
- Um Kurven zu überlagern, müssen Sie `hold on` nach dem ersten `plot`-Befehl aufrufen.
- Sie können Plot-Fenstern eine Überschrift geben mit z.B. `title('Histogramm');`

- c) Überlagern Sie zusätzlich das Histogramm der Intensitäten (Helligkeiten). Nützlich zur Berechnung der Intensitäten ist eine Funktion von dem vorigen Aufgabenblatt.

Wird das Bild (ohne es bisher angezeigt zu haben) wohl eher zu dunkel oder zu hell dargestellt werden?

Übernehmen Sie das Histogramm und Ihre Vermutung in das Protokoll.

Lassen Sie sich das Bild mit `imshow` anzeigen. Hatten Sie mit Ihrer Vermutung recht?

Optional: Lassen Sie sich mit `showQuadView` wie bei dem vorigen Aufgabenblatt die RGB- und HSI-Kanäle kombiniert und einzeln anzeigen.

- d) Berechnen Sie das Minimum, das Maximum und den globalen Kontrast der Bildhelligkeit (Intensität) mit Hilfe der Funktionen `min` und `max`.

Beachten Sie dabei: In MATLAB arbeiten viele Berechnungsfunktionen (darunter `min`, `max`) entlang jeder Spalte. Sie bekommen also das Minimum/Maximum je Spalte. Mit `array(:)` wird aus jedem `array` (egal welcher Dimension) ein eindimensionales Array mit nur einer Spalte.

Übernehmen Sie den Kontrastwert in das Protokoll.

- e) Berechnen Sie nun einen stabilen, rauschunempfindlichen Wert für den globalen Kontrast mit der Funktion `stretchlim`. In der Dokumentation (`doc stretchlim`) finden Sie Details dazu. Übernehmen Sie auch diesen, stabilen Kontrastwert in das Protokoll.

- f) Der stabile Wert für den Kontrast unterscheidet sich recht deutlich von dem mit Hilfe der Funktionen `min` und `max` berechneten Kontrast. Um dies nachzuvollziehen, lassen Sie sich die Zahlenwerte des Histogramms der Intensitäten aus Teilaufgabe c ausgeben. Ist dies auch an dem geplotteten Histogramm erkennbar?

- g) Berechnen Sie die Entropie des Bildes.

2 Kontrastverbesserung von Bildern

Ziel dieser Aufgabe ist es, den Kontrast des Bildes zu verbessern. Hierbei bitte *keine* speziellen MATLAB-Funktionen (z.B. nicht `imadjust`) verwenden, sondern die Formel selbst implementieren.

Setzen Sie nach der Kontrastverbesserung jeweils wieder das Farbbild so zusammen, dass es als RGB-Bild angezeigt werden kann und übernehmen Sie es in die Ausarbeitung. Das Bild sollte also jeweils (etwas) besser aussehen, als das Originalbild.

- a) Führen Sie eine Histogrammspreizung der Werte des I-Kanals des Bildes mit den Werten, die von `stretchlim` geliefert werden, durch.
- b) Führen Sie eine Gamma-Korrektur der Werte des I-Kanals des Bildes durch. Probieren Sie verschiedene Werte für den Parameter γ und wählen Sie einen Ihnen geeignet erscheinenden Wert. Geben Sie den gewählten Wert in der Auswertung an.

- c) Führen Sie eine Histogrammlinearisation (engl. histogram equalization) der Werte des I-Kanals des Bildes mit der MATLAB-Funktion `histeq(I)` und `adapthisteq` durch.

Optional: Probieren Sie dies (oder alles) auch mit dem Bild `AquaTermi_lowcontrast` aus.

- d) Berechnen Sie die Entropie des Bildes nach der Gamma-Korrektur und der Histogrammlinearisation und vergleichen Sie sie mit dem Wert vor der Korrektur.

- e) Optional: Lassen Sie sich die Histogramme des I-Kanals vor und nach den drei verschiedenen Kontrastkorrekturen mit der Funktion `imhist(I)` anzeigen. Die Anzahl der verwendeten Intervalle lässt sich einstellen: `imhist(I,n)`. Testen Sie auch `n=10`. Übernehmen Sie je ein Histogramm in die Auswertung.