



## Aufgabe 5: Hough Transformation

Implementieren Sie die Hough-Transformation für Geraden. Dazu soll das Bild mit Canny-Edge vor-verarbeitet werden und das Ergebnis des Canny-Algorithmus als Eingabe für die Geraden-Detektion dienen (Aufgabe 4a ist also Voraussetzung für die Bearbeitung).

- Nutzen Sie die Geraden-Darstellung in HNF um eine Geradendetektion mit der Hough Transformation zu realisieren. Zeichnen Sie das Akkumulator-Array in ein zusätzliches Graustufen-Bild und codieren Sie den Wert der Zellen mit Helligkeiten.
- Implementieren Sie die verschiedenen vorgestellten Varianten wie das Akkumulator-Array ausgewertet werden kann und zeichnen Sie die Ergebnis-Geraden in das Ursprungsbild. Die verwendeten Schwellwerte sind per GUI einstellbar.
- Erweitern Sie die Implementierung um die Nutzung der Kantenstärke und der Berücksichtigung der möglichen Länge einer Geraden.
- Erweitern Sie die Implementierung so, dass auch Geraden-Segmente erkannt werden können. Erweitern Sie dazu das Akkumulator-Array so, dass die Bounding-Box der Geraden beim Hochzählen der Einträge entsprechend angepasst wird.

Die Implementierung erfolgt im Verzeichnis 'Sheet4' in der Datei `hough.cpp`. Folgende Methoden sind für die jeweilige Funktionalität vorgegeben und sollen mit Inhalt gefüllt werden:

- `QImage* showAccumulatorArray(QImage* image, int angular_steps, int radial_steps, bool do_bias_compensation, double threshold_a_min, bool do_threshold_reduction, bool do_non_maxim_suppression, double canny_sigma, int canny_tHi, int canny_tLo);`
- `QImage* drawHoughLinesInImage(QImage* image, int angular_steps, int radial_steps, bool do_bias_compensation, double threshold_a_min, double canny_sigma, int canny_tHi, int canny_tLo);`

Die Bedeutung und Wertebereiche der Übergabeparameter ist im Code oberhalb des Methodenrumpfes dokumentiert.