



HOCHSCHULE KONSTANZ TECHNIK, WIRTSCHAFT UND GESTALTUNG
UNIVERSITY OF APPLIED SCIENCES

**Technische Grundlagen
der angewandten Informatik**

Versuch 2

Benedict Roth, David Kubatzki

Konstanz, 29. April 2015

Zusammenfassung (Abstract)

Thema:	Versuch 2	
Autoren:	Benedict Roth	beroth@htwg-konstanz.de
	David Kubatzki	dakuba@htwg-konstanz.de
Betreuer:	Prof. Dr. Matthias O. Franz	mfranz@htwg-konstanz.de
	Jürgen Keppler	juergen.keppler@htwg-konstanz.de
	Martin Miller	martin.miller@htwg-konstanz.de

«Zusammenfassung etwa 100 Worte.»

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Listingverzeichnis

Abkürzungsverzeichnis

1

Einleitung

In diesem Versuch untersuchen wir die Eigenschaften einer Digitalen Kamera, in unserem Fall eine Webcam der Firma Logitech. Wir werden diese Kalibrieren. Dies geschieht nach dem Verfahren, wie auch bei industriellen Inspektionsanlagen, bei der Fernerkundung durch Satelliten oder in der Astronomie. Zuerst nehmen wir einen Graukeil der verschiedene Graustufen abbildet, parallel zur Kamera auf. Danach wird der Graukeil am PC in seine Graustufen aufgeteilt. Als nächstes nehmen wir ein Dunkelbild auf, damit wir unterschiedliche Nullpunkte, welche manche Pixel haben ausgleichen können. Danach nehmen wir ein Weißbild auf, um die unterschiedliche linearität der Beleuchtungsstärke der einzelnen Pixel zu kompensieren. Zum schluss sollen noch so genannte dead pixel auf dem Weißbild und sogenannte Stuck und hot Pixel auf dem Schwarzbild gekennzeichnet werden.

2

Versuch 1

2.1 Fragestellung, Messprinzip, Aufbau, Messmittel

Zuerst haben wir geschaut, wie gut die Webcam einen Grauwertverlauf aufnimmt und können, da das Muster bekannt ist, die Wiedergabequalität des Kamerasensors messen.

2.1.1 Messmittel

Als Messmittel dient eine Webcam vom Modell "c270" der Firma Logitech, und hat eine Auflösung von 720p, was einer Auflösung von 1280x720 Pixeln entspricht. Dies sind 921600 Bildpunkte.

2.1.2 Messprinzip

Der Sensor ist ein integrierter Schaltkreis, der aus einer Matrix von mehreren Millionen von Fotozellen besteht. Jede dieser Fotozellen stellt einen Pixel des Sensors dar und wandelt das einfallende Licht auf Grund des inneren photoelektrischen Effekts in elektrische Ladung um. Ein Pixel selbst besteht aus einer Mikrolinse, einem Farbfilter, einer Reihe von Leiterbahnen, zwei Elektroden, einer isolierenden Oxidschicht und dotiertem Silizium. Die Mikrolinse bündelt das Licht, so dass möglichst viel Licht auf den lichtempfindlichen Teil des Pixels treffen kann. Photodioden reagieren auf Licht in einem breiten Farbspektrum. Mit ihnen kann man die Helligkeit messen und in elektrische Information umwandeln. Um Farbinformationen zu erhalten, ist vor jedem Pixel ein Farbfilter angebracht, welcher nur Licht eines bestimmten Spektralbereichs passieren lässt. Aus der Farbinformation benachbarter Pixel kann die Kamerasoftware sich dann die Farben eines jeden Pixels errechnen.

2.1.3 Aufbau

Die Kamera ist an einem Ständer befestigt, an welchem die Neigung der Kamera sowie die Höhe durch ein Rad eingestellt. Die Kamera ist somit parallel zur Tischoberfläche positioniert. Der Graukeil ist so ausgerichtet, dass er möglichst das gesamte Bild der Kamera ausfüllt. Die Kamera wird per USB an den Laborrechner angeschlossen. Das aufzunehmende Objekt ist ein Graukeil. Ein Graukeil ist eine Platte, die fünf verschiedene Graustufen aufgedruckt hat.

2.2 Messwerte

Bevor wir mit dem Aufnehmen des Bildes beginnen, haben wir die Einstellungen der Kamera ausgelesen und abgespeichert. Dies funktioniert mit dem einem Code Listing, welches in einem Einführungsskript zum Modul OpenCV von der Lernplattform Moodle, welches wir auch benutzen. Zum Aufnehmen des Weißbildes, haben wir ebenfalls darauf das Skript für das Anzeigen von Live-Bildern einer Webcam übernommen. Dieses Skript haben wir dann um den Befehl `cv2.write(BILDNAME,BILDQUELLE)` erweitert so dass ein Bild aufgenommen wird, nachdem man die q Taste bei dem angezeigten Vorschaubild drückt. Das Bild wird dabei wie in der Versuchsanleitung gefordert im verlustfreien png Format gespeichert, da das JPEG Format verlustbehaftet ist. Anzumerken sei noch, dass gefordert war das Das Bild von einem Farbbild in ein Graubild umgewandelt werden soll. Glücklicherweise waren die Standarteinstellungen schon so eingestellt, dass ein Graubild aufgenommen wurde.

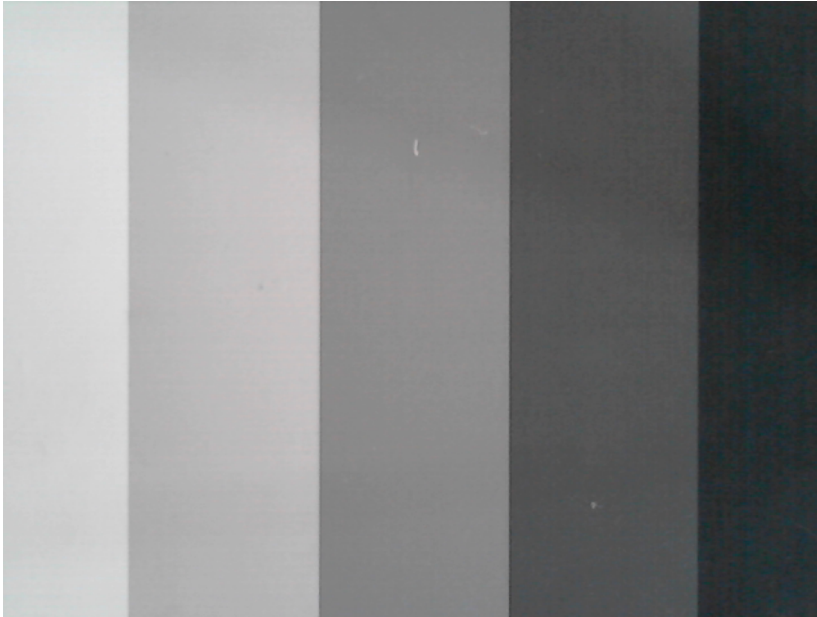


Abbildung 2.1: Graukeilbild

2.3 Auswertung

Als nächstes müssen wir das Bild wieder einlesen und in seine Farbstufen aufteilen. Das einlesen funktioniert hierbei über die Funktion `cv2.imread(BILDNAME)`, was uns ein Array aus einzelnen Pixelwerten zurückgibt. Um nun die einzelnen Graustufen des Graukeils in einzelne Farben aufzuteilen, nutzen wir das Index Slicing von python. Wir achten hierbei darauf, dass wir nicht die einzelnen Ränder mit in die Bilder nehmen haben wir jeden Abschnitt 90 pixel breit gemacht und so hoch wie das Ursprungsbild.

Für jede Stufe muss jetzt noch der Mittelwert bestimmt werden. Hierzu wenden wir die Funktion `np.mean(STUFE)` aus dem Numpy Modul auf die jeweilige Stufe an. Dazu berechnen wir ebenfalls die Standardabweichung mithilfe der Funktion `np.std(STUFE)` Anschließend haben wir die Einzelnen Graustufen abgespeichert. Unterschieden haben wir hier die Fünf Stufen Weiß, Hellgrau, Grau, Dunkelgrau und Schwarz.

2.4 Interpretation

Wir haben jetzt die Mittelwerte und die Standardabweichungen berechnet um diese nach der Kalibrierung des Sensors als Vergleich heranziehen zu können.

3

Versuch 2

3.1 Fragestellung, Messprinzip, Aufbau, Messmittel

3.1.1 Fragestellung

Im zweiten Teil des Versuchs, erstellen wir 10 Dunkelbilder. Ein Dunkelbild ist ein Bild bei kompletter Verdunklung des Sensors. Dieses brauchen wir, weil nicht jeder Pixel den Grauwert 0 hat wenn der Sensor verdeckt wird. Das liegt am thermischen Rauschen der Ausleseelektronik und am Dunkelstrom. So müssen wir ein Dunkelbild erstellen um diesen Effekt aus normalen Bildern zu eliminieren. Anzumerken sei noch, dass für jede unterschiedliche Belichtungszeit ein eigenes Dunkelbild benötigt wird.

3.1.2 Aufbau

Am Aufbau selbst hat sich nicht viel geändert. Nun brauchen wir noch zusätzlich ein Objekt, um den Sensor zu verdecken.

3.1.3 Messmittel

Als Messmittel dient wieder die Webcam aus Versuch 1.

3.2 Messwerte

Zur Aufnahme des Dunkelbildes müssen wir nun den Sensor abdecken. Zur Aufnahme modifizieren wir nun das Python Skript aus Aufgabe 1 und bearbeiten hier den Teil welcher

für das Aufnehmen des Bildes zuständig ist. So benutzen wir hier nun eine Schleife in der im Intervall von 0 bis 10 jeweils in 1er schritten hochgezählt wird und jeweils die Funktion `cv2.(BILDNAME+ZÄHLER+FORMAT,BILDQUELLE)` aufgerufen und können so einfach die Bilder von 0 bis 10 durchnummerieren und abspeichern. Belichtungsparameter waren bei der Aufnahme die Gleichen wie bei Versuch1, da nichts verändert wurde. Auch hier ist wieder zu beachten, dass wir aufgrund der Voreinstellungen gleich beim Abspeichern Graubilder erhalten haben.

3.3 Auswertung

Als nächstes müssen wir die 10 Bilder wieder einlesen, dies machen wir mithilfe einer Schleife nach dem Schema `img = cv2.imread('dbild'+str(ZÄHLER) + '.png')` und können so bequem alle 10 Bilder einlesen. Bei jedem Durchgang werden die Bilder noch aufaddiert um anschließend durch die Menge der Bilder geteilt zu werden um so den Mittelwert bilden zu können. Um eine kontrastmaximierte Darstellung zu erhalten, wird nun das Dunkelbild mit 100 multipliziert. In den nächsten Versuchen benötigen wir jedoch nur das normale Dunkelbild. Anschließend schreiben wir uns noch ein Python Skript, welches das Dunkelbild wieder einliest und dieses vom ursprünglichen Graukeilbild subtrahiert. Hierzu subtrahiert man einfach die beiden Bilddateien die man beim Einlesen erhält voneinander.

3.4 Interpretation

Wir haben nun ein Dunkelbild erstellt um die Stuck und Hotpixel ausfindig machen und diese zu eliminieren.



Abbildung 3.1: Graukeilbild

4

Versuch 3

4.1 Fragestellung, Messprinzip, Aufbau, Messmittel

Die einzelnen Pixel haben zwar eine hervorragende Linearität mit der Beleuchtung, aber aufgrund von Fertigungstoleranzen ist ihre Sensitivität nicht gleich. Außerdem tritt noch die so genannte Vignettierung auf. Hierbei wird die Helligkeit nicht gleichmäßig auf den Sensor von der Optik übertragen. Dies äußert sich in einer Abdunklung des Bildes zu den Rändern hin. Um diesen Effekt zu kompensieren, nimmt man ein so genanntes Weißbild auf durch welches das normale Bild dividiert wird. Zu beachten ist noch, dass das Weißbild vom Fokus des Kamerasensors abhängt.

4.1.1 Aufbau

Aufbau ist der Selbe wie bisher, bloß dass wir diesmal eine homogene Fläche Brauchen, diese ist in unserem Fall ein weißes Blatt.

4.2 Messwerte

Das aufnehmen der Weißbilder funktioniert ebenfalls wie in Aufgabe 2, mit dem Unterschied, dass wir diesmal anstatt den Sensor zu verdecken, ein weißes Blatt Papier, im gleichen Abstand wie den Graukeil ablichten. Zusätzlich haben wir die Belichtung auf ca. 40% der Hellsättigung gesetzt.

4.3 Auswertung

Nun lesen wir ebenfalls wieder die Bilder ein und bilden den Mittelwert mit dem gleichen Verfahren wie schon in Aufgabe 2. Dadurch erhalten wir ein Weißbild. Das Skript zur Korrektur des Graukeils wird nun so erweitert, dass das Weißbild ebenfalls eingelesen wird. Dieses wird nun in Double umgewandelt und anschließend normiert. Die Normierung erfolgt über die Berechnung des Mittelwertes durch den das Bild dividiert wird.

4.4 Interpretation

Nun haben wir ein Weißbild erstellt auf dem man dead pixel erkennen kann, welche wir in unserem Fall nicht haben. Das python skript zur Korrektur des Graukeils wurde so erweitert, dass nicht nur stuck und hot pixel sondern auch dead pixel eliminiert werden.

5

Versuch 4

5.1 Fragestellung, Messprinzip, Aufbau, Messmittel

5.2 Messwerte

5.3 Auswertung

5.4 Interpretation

Anhang

A.1 Quellcode

A.1.1 Quellcode Versuch 1

A.1.2 Quellcode Versuch 2

A.1.3 Quellcode Versuch 3

A.1.4 Quellcode Versuch 4

A.2 Messergebnisse

[?, S.21]