

Medizinische Bildverarbeitung - Optical Imaging

Eldin Ramic, Alexander Straube
Hochschule München
München, Deutschland
e.ramic@hm.edu, straube@hm.edu

I. AUFGABENSTELLUNG

Ziel der Aufgabe ist es die optischen Bilddaten eines flexiblen Endoskops einzulesen, aufzubereiten und darzustellen. Die Bilddaten bestehen aus einem Farb- und einem korrespondierenden Fluoreszenzbild. Zur Lösung der Aufgabe soll selbständig geeigneter Quellcode erstellt werden. Die Programmiersprache Python wurde zur Lösung aller Teilaufgaben verwendet.

II. DATENSATZ

Die Bilddaten liegen im FITS Dateisystem vor (<http://fits.gsfc.nasa.gov/>). Dieses Bilddateisystem ist in der Wissenschaft vor allem in der Astronomie verbreitet und zeichnet sich durch große Vielseitigkeit aus (Meta Informationen, hochdimensionale Datenwürfel...).

III. EINFACHE ÜBERLAGERUNG

Hierbei sollen die Farbbilder und die Fluoreszenzbilder überlagert werden. Diese müssen zunächst eingelesen und gemäß der Zeitstempel zeitlich zugeordnet werden. Die Zeitstempel der Bildaufnahmen lassen sich aus den Metadaten der Bilder entnehmen. Anschließend werden die Bilder mithilfe der gespeicherten Registrierungsparametern (aus den Metadaten) koregistriert.

A. Einlesen der Bilddaten

Zum Einlesen der Bilddaten wird das Modul *fits* aus der Bibliothek *astropy.io* verwendet.

B. Zeitliche Zuordnung

Die Zeitstempel der einzelnen Bilder werden aus den Metadaten herausgelesen. Für die zeitliche Zuordnung wird das folgende Verfahren angewendet:

- Lese Zeitstempel aus den Metadaten des jeweiligen Farbbildes I_n ab.
- Finde das Fluoreszenzbild mit der geringsten absoluten Differenz zwischen dem Zeitstempel des Fluoreszenzbildes und des Farbbildes I_n .
- Wiederhole das Verfahren für alle weiteren Farbbilder

C. Koregistrierung

Vor der Transformation werden die Fluoreszenznilder auf die Größe der Farbbilder skaliert. Die jeweils neun Registrierungsparameter werden aus den Metadaten herausgelesen (TRANSC0, ..., TRANSC8).

Dadurch ergibt sich die Transformationsmatrix M:

$$M = \begin{bmatrix} \text{TRANSC0} & \text{TRANSC1} & \text{TRANSC2} \\ \text{TRANSC3} & \text{TRANSC4} & \text{TRANSC5} \\ \text{TRANSC6} & \text{TRANSC7} & \text{TRANSC8} \end{bmatrix}$$

Zusätzlich werden die Fluoreszenzbilder binarisiert und die einzelnen Ausreißer durch eine mehrstufige Erosion entfernt.

Ein Beispiel für ein korrespondierendes Bildpaar:

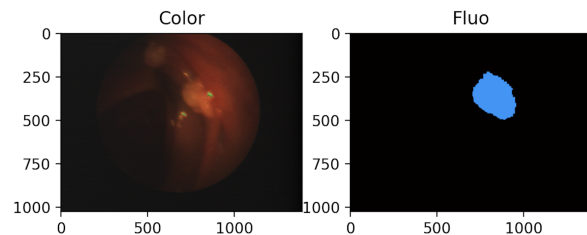


Abbildung 1. links: Farbbild; rechts: Fluoreszenzbild

D. Überlagerung

Ein Beispiel für eine Überlagerung eines korrespondierenden Bildpaares:

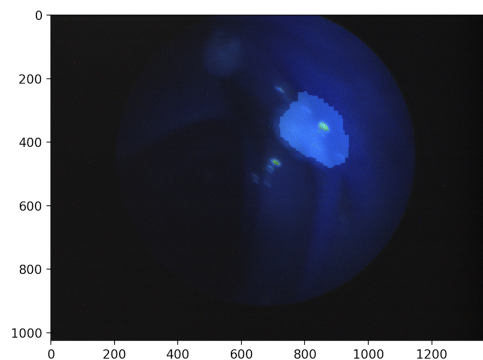


Abbildung 2. Überlagertes Bild

IV. ERSTELLEN EINES HOCHAUFGELÖSTEN ÜBERLAGERTEN FALSCHFARBENVIDEOS

Hierbei sollen die Reflektionsdaten aus einer Videodatei (.mp4) eingelesen und mit den korrespondierenden Fluoreszenzbildern überlagert werden. Eine geeignete adaptive Methode zur Bestimmung der Registrierungsparameter (in Ort und Zeit) ist hier nötig.

Idee: Bestimme die zugehörige Konturen und berechne anhand der Kontur-Mittelpunkte die Transformationsmatrix

A. Vorverarbeitung

Da der Bilddatensatz der Reflektionsdaten deutlich größer ist als der der Fluoreszenzbildern, muss dieser zunächst angepasst werden. Dabei werden einzelne Fluoreszenzbilder mehrfach dem Bilddatensatz beigefügt, sodass die Größen der Bilddatensätze übereinstimmen.

Bevor die Konturen berechnet werden, wird die Auflösung beider Bilddatensätze zur Reduzierung des Berechnungsaufwandes auf 480x480 Pixel reduziert. Zudem werden auch die Reflektionsbilder binarisiert, um die Anzahl der falschen Konturen zu reduzieren.

B. Ansatz

Das Ziel ist, die passende Kontur in beiden korrespondierenden Bildern zu bestimmen und den Kontur-Mittelpunkt zu berechnen. Denn mithilfe der Mittelpunktskordinaten kann die jeweilige Transformationsmatrix bestimmt werden. In diesem Fall wird eine Affine-Transformation durchgeführt. Da die Größen der Konturen ungefähr übereinstimmen, wird mit der Transformation lediglich die Kontur anhand des Zentrums in Richtung der x- und y-Achse verschoben. Die Verschiebungen werden wie folgt berechnet:

(x_s/y_s) : Verschiebung

(x_1/y_1) : Mittelpunkt (Reflexionsdaten)

(x_2/y_2) : Mittelpunkt (Fluoreszenzbilder)

$$x_s = x_1 - x_2$$

$$y_s = y_1 - y_2$$

C. Weitere Maßnahmen

Des Weiteren werden folgende Maßnahmen zur Detektion der korrekten Kontur innerhalb der Reflektionsbilder getroffen:

- **Bereichseingrenzung:** Im gesamten Bilddatensatz lässt sich das Bild auf die linke-obere Bildhälfte reduzieren, da die korrekte Kontur nur innerhalb dieser sich befindet.

- **Iterative Helligkeitserhöhung:** In einigen Fällen werden alle Konturen heraus-binarisiert. Dies kommt vor, wenn das Bild vergleichsweise dunkel ist und die Binarisierung in

der Vorverarbeitung nicht optimal gewählt ist. Diesem wird entgegengesteuert, indem die Helligkeit iterativ erhöht wird, bis eine Kontur ermittelt werden kann.

D. Ergebnis

In den meisten Fällen werden die Bilder korrekt überlagert. Nichtsdestotrotz kommt es an manchen Stellen vor, dass die falsche Kontur als 'einzige' Kontur bestimmt wird und somit die Transformation nicht korrekt durchgeführt wird. Außerdem hat sich Bereichseingrenzung als positiv erwiesen, da dabei immer die korrekte Kontur zur Bestimmung der Transformationsmatrix gewählt wurde.