

Beleg 2: Generierung einer Ground Truth

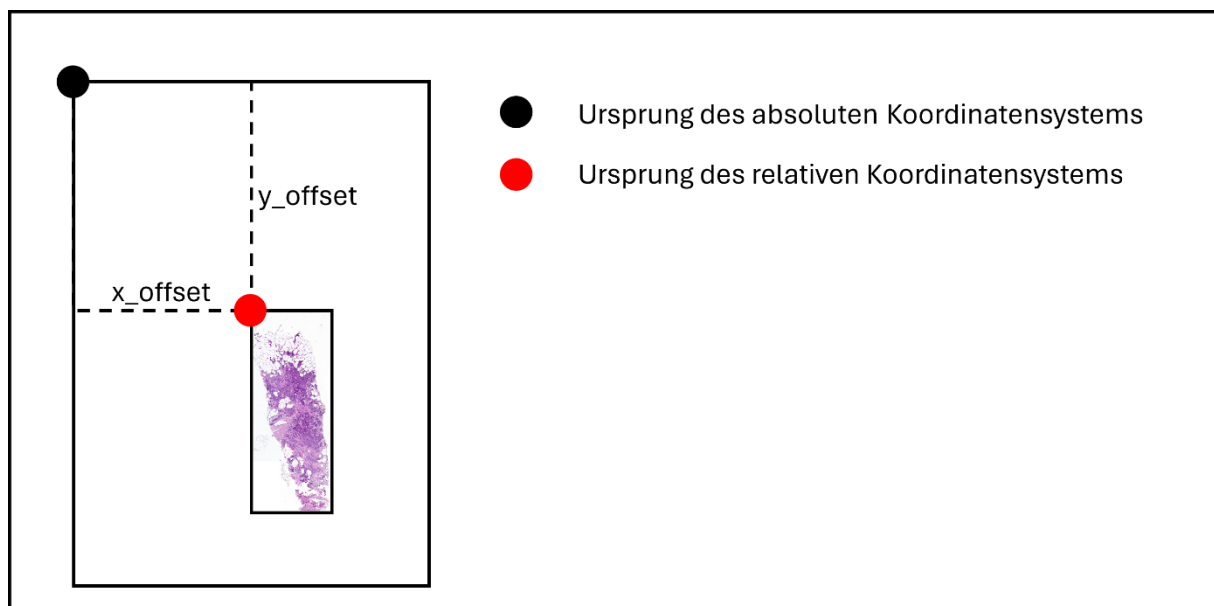
Achtung: Die Daten dürfen nicht veröffentlicht werden.

In diesem Beleg geht es darum, aus bestehenden Bilddaten und Annotationen eine Ground Truth zum Trainieren eines neuronalen Netzes zur Detektion von Mitosen zu generieren.

Zu diesem Zweck wird ein Bild (WSI.jpg) sowie zwei Dateien (annotations.csv und offset.csv) zur Verfügung gestellt. In annotations.txt finden Sie zwei Spalten: Die ID der jeweiligen Annotation sowie die Koordinaten. Diese wurden algorithmisch generiert und liegen als float vor. Beachten Sie also, dass diese nicht eins-zu-eins für die Indizierung einzelner Pixel genutzt werden können.

Neuronale Netze erwarten in der Regel recht kleine Bildkacheln in einer festgelegten Größe. Die Ground Truth wird also aus einer Menge an Bildkacheln sowie jeweils zugehörigen Segmentierungen bestehen. Für diesen Beleg nutzen Sie eine Kachelgröße von 224.

Die Koordinaten in „annotations.csv“ beschreiben die Mittelpunkte der Mitosen. Das WSI wurde mit einem 3DHitech Scanner digitalisiert, der ein globales Koordinatensystem für den gesamten histologischen Schnitt nutzt, als Bilddaten jedoch nur die Bounding Box um das Gewebe generiert. Der Koordinatenursprung (0, 0) des Bildes entspricht daher nicht dem Koordinatenursprung der Annotationen, da sich diese auf die globalen Koordinaten beziehen. Die beiden Werte in offset.csv beschreiben die relativen Koordinaten des Bildes im globalen Koordinatensystem. Folgende Grafik visualisiert diesen Sachverhalt.

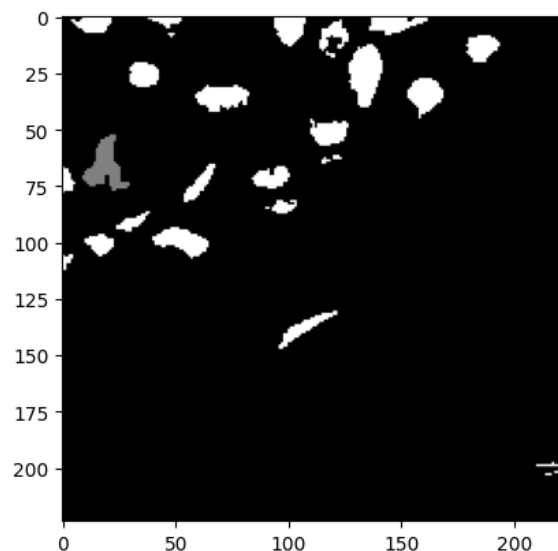


Erstellen Sie nun ein neues Jupyter Notebook und bearbeiten Sie die folgenden Aufgabestellungen:

Aufgabe 1: Schreiben Sie einen Algorithmus, mit dem Sie das Bild zunächst in einzelne Kacheln zerlegen können. Dabei sollte es möglich sein mit anzugeben, wie viele Pixel sich zwischen jeweils benachbarten Kacheln überlappen (der Wert sollte in Richtung der x- und y-Achse gleich sein, es reicht also ein Parameter). Stellen Sie mittels geeigneter Vorverarbeitung (bspw. durch eine einfache Segmentierung) sicher, dass nur Kacheln generiert werden, in denen auch Gewebe zu sehen ist.

Aufgabe 2: Schreiben Sie einen Algorithmus zur Segmentierung von Zellkernen innerhalb der jeweiligen Kacheln. Sie sollten hierzu eine geeignete Vorverarbeitung wählen, um sicherzustellen, dass wirklich nur Zellkerne segmentiert werden. Hierzu ist es empfehlenswert, immer wieder visuell zu überprüfen, ob Ihr Algorithmus auch korrekt arbeitet – idealerweise auch in Kacheln, in denen keine Zellkerne enthalten sind.

Aufgabe 3: Überarbeiten Sie ihre Segmentierungsmasken dahingehend, dass alle Mitosen eine separate Klasse (sprich einen vom Hintergrund und den anderen Zellkern verschiedenen Grauwert in der Segmentierungsmaske) bekommen. Die globalen Positionen der Mitosen können Sie anhand der Koordinaten in annotations.txt bestimmen. Sie müssen also eine Logik implementieren, anhand der Sie die globalen Koordinaten auf das lokale Koordinaten des Bildes bzw. ihrer Kacheln abbilden können. Visualisieren Sie anschließend alle Kacheln mit Mitosen sowohl als Originalbild als auch als Segmentierungsmaske (mit den Klassen für Hintergrund, Zellkern und Mitose). Folgende Grafik zeigt ein Beispiel für eine korrekte Segmentierung (Zellkerne in Weiß, die Mitose bei ca. $x=65$, $y=30$ in Grau) aus dem letzten Semester.



Noch ein Tipp: Ich rate Ihnen, für die Zuordnung der Annotationen zu den jeweiligen Kacheln objektorientiert vorzugehen. Das kann die Komplexität Ihres Codes deutlich reduzieren.