



**Technische Hochschule
Brandenburg**
University of
Applied Sciences

Dokumentation

Projekt Data Mining

Image Segmentation

Autor: Sebastian Fabig
Matrikelnummer: 20120053
Email: fabig@th-brandenburg.de

Inhaltsverzeichnis

1 Aufgabenstellung	1
2 Datenbeschreibung	2
3 U-Net - Convolutional Neural Network	4
4 Verwendetes U-Net	6
4.1 Implementierung	6
4.2 Unterschiede zum Original	8
5 Experimente	9
5.1 Experiment 2	9
5.1.1 Kreuzvalidierung	9
5.1.2 Ergebnis	10
5.2 Experiment 3	12
5.2.1 Wichtungsvergleich	12
5.2.2 Ergebnis	13
6 Zusammenfassung	15

1 Aufgabenstellung

Aufgabe des Projekts ist es, ein zur Segmentierung von Radiologieaufnahmen der Lunge trainiertes DeepLearning-Netz zu untersuchen und auf die vorgegebene Radiologieaufnahmen der Prostata anzuwenden. Dabei soll die Struktur des Netzes untersucht werden. Ebenso wird untersucht, inwieweit sich das vortrainierte Netz auf die gegebenen Prostatabilder umtrainieren lässt.

2 Datenbeschreibung

Die Daten umfassen 354 Radiologieaufnahmen der Prostata mit einer entsprechenden Binärmaske. Die Masken wurden von Radiologen manuell erstellt. 284 Aufnahme-Maske-Paare werden als Trainingsmenge verwendet. Die restlichen 70 Paare werden als Testmenge verwendet. Sowohl Aufnahmen als auch Masken haben eine Abmessung von 400x400.

Die Aufnahmen sind Graustufenbilder im JPG-Format. Abbildung 2.1 zeigt eine Beispieldatenaufnahme.



Abbildung 2.1: Beispiel - Prostataaufnahme

Die Masken sind Binärbilder im PNG-Format. Abbildung 2.2 zeigt eine Beispielmaske.

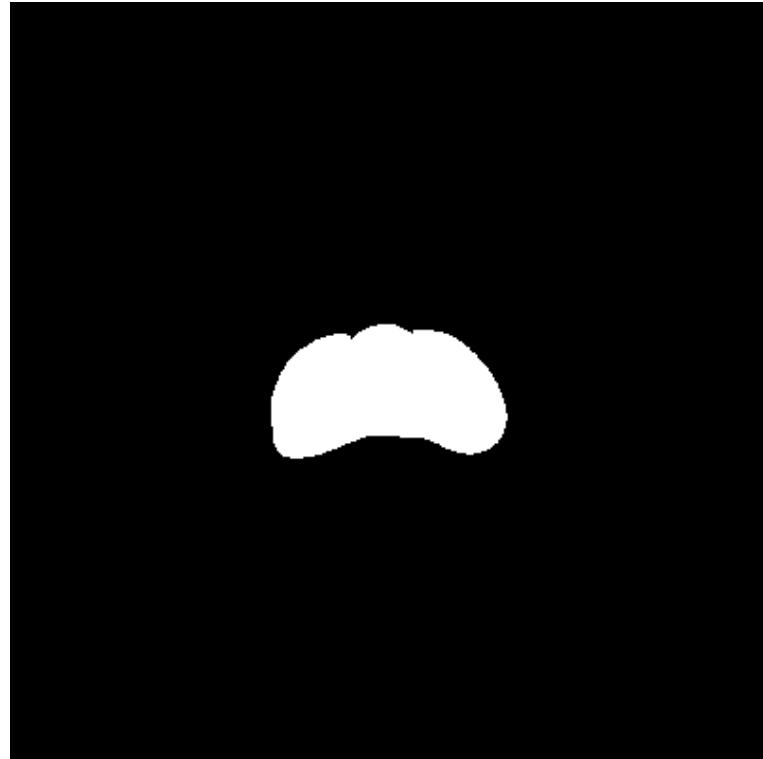


Abbildung 2.2: Beispiel - Binärmaske

3 U-Net - Convolutional Neural Network

Das U-Net ist ein Convolutional Neural Network zur Segmentierung von medizinischen Bildern. Es wurde 2015 von Olaf Ronneberger, Philipp Fischer und Thomas Brox von der Albert-Ludwigs-Universität Freiburg vorgestellt. Zuvor gewannen sie mehrere Challenges mit der Anwendung des U-Nets [Ron+15]. Die Architektur ihres verwendeten Netzes ist in Abbildung 3.1 zu sehen.

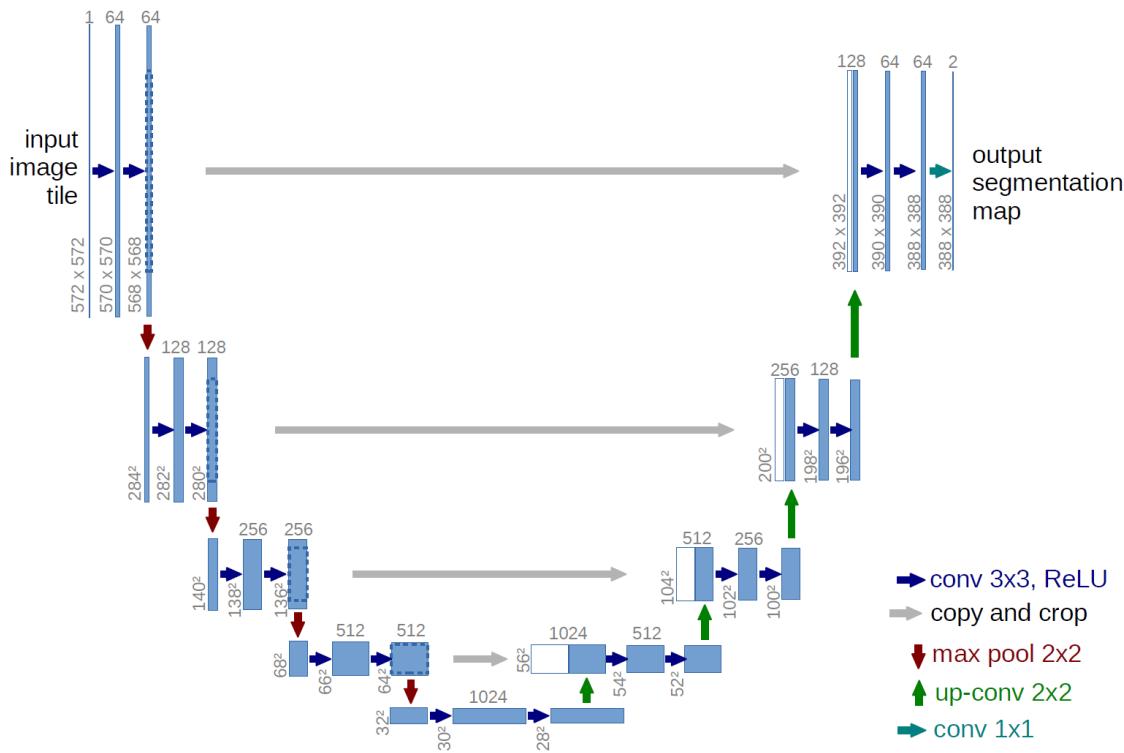


Abbildung 3.1: U-Net Architektur [RFB15]

Das U-Net basiert auf der Struktur eines Auto-Encoder und besteht aus 23 Layer. Als Fehlerfunktion wird im Original die Kreuzentropie verwendet. Wie diese besitzt es einen Encoder und einen Decoder. Als Input des U-Nets wird ein Bild mit einem einzigen Feature-Channel erwartet. Der Encoder besteht aus zwei 3×3 Convolutional Layer mit einer ReLU-Aktivierungsfunktion. Anschließend erfolgt ein Max-Pooling mit einem 2×2 -Filter und einer

Schrittgröße von 2. Dies sorgt für eine Informationsreduktion, da nur die stärkste Aktivierung jedes 2x2-Quadrats aus Neuronen des Convolutional Layers übernommen wird. Diese Schritte werden vier mal wiederholt. Dabei wird nach jedem Max-Pooling die Anzahl der vorhandenen Feature-Channels verdoppelt.

Im Decoder erfolgt ein Upsampling der Feature-Map mit einer anschließenden 2x2-Faltung, wodurch die Anzahl der Feature Channels halbiert wird. Anschließend wird diese Feature-Map mit der entsprechenden Feature-Map des Encoders konkateniert. Abschließend folgen erneut zwei 3x3 Convolutional Layer mit einer ReLU-Aktivierung. Als letztes Layer wird eine 1x1-Faltung verwendet, um den erstellten Feature-Vektor auf die gewünschten Klassen zu mappen. Als Output gibt das U-Net eine Segmentierungs-Map zurück. (vgl. [RFB15])

4 Verwendetes U-Net

Im Rahmen des Projektes wurde ein öffentliches U-Net-Projekt auf GitHub verwendet. In diesem wurde eine dem U-Net ähnliche Netzarchitektur verwendet. Das Netz wurde zur Segmentierung der Lungenflügel in Radiologieaufnahmen des Brustkorbes verwendet und trainiert.

4.1 Implementierung

Zur Implementierung wurden Keras 2.0.4 mit TensorFlow 1.1.0 als Backend verwendet. Zum Training des Netzes wurden das *Montgomery County X-ray Set* der *United States National Library of Medicine* sowie Datensätze der *Japanese Society of Radiological Technology* verwendet. Die Masken für den letztgenannten Datensatz wurden vom *Image Sciences Institute* der *University Medical Center Utrecht* erstellt. Zur Datenaugmentierung wird der in Keras implementierte ImageDataGenerator mit leichten Modifizierungen verwendet. Als Optimierungsalgorithmus wird Adam verwendet. Adam ("adaptive moment estimation") ist ein alternativer Optimierer zum klassischen Gradientenabstieg und wird heute generell für Deep-Learning-Probleme verwendet. Als Fehlerfunktion wird die binäre Kreuzentropie verwendet. Als Gütemaße werden der Jaccard-Koeffizient beziehungsweise "Intersection over Union" (IoU) und der Sorensen-Dice-Koeffizient verwendet.

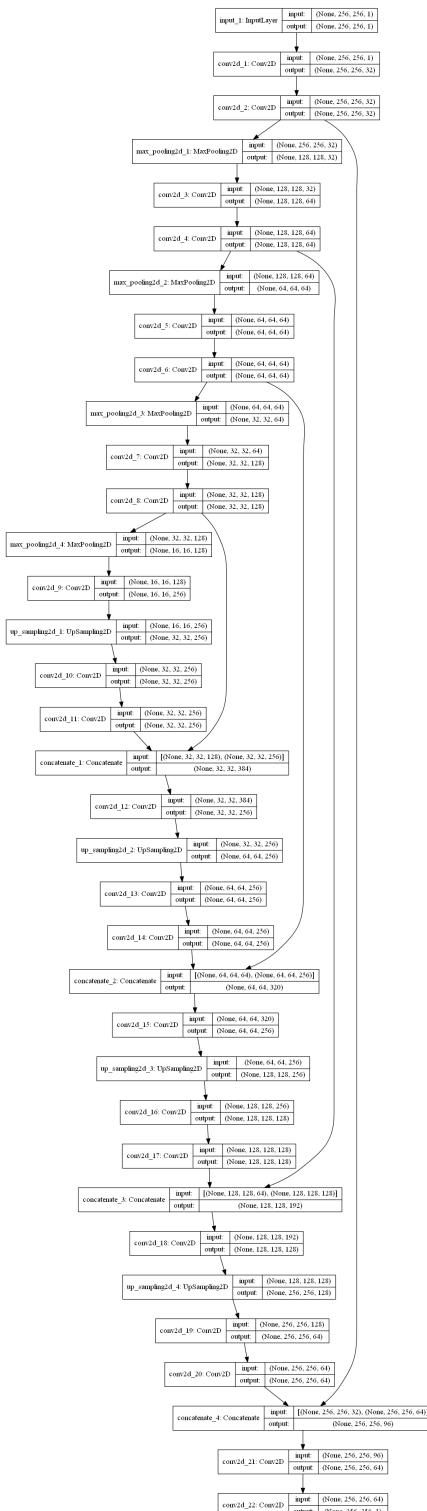


Abbildung 4.1: Keras-Modell [Pet17]

4.2 Unterschiede zum Original

Abbildung 4.1 zeigt die Architektur des U-Nets. Im Unterschied zum Original werden 22 statt 23 Convolutional Layer verwendet. Das fehlende Layer befindet sich in der Mitte des Netzes beziehungsweise am "Boden des Us". Nach dem letzten Max-Pooling wird nur ein statt zwei 3x3 Convolutional Layer verwendet (vgl. conv2d_9 in Abbildung 4.1). Ein weiterer Unterschied zum Original ist die Konkatenation der Feature-Maps. Die erfolgt in der GitHub-Implementierung erst nach den 3x3-Faltungen, die dem Upsampling folgen. Die der Konkatenation folgende Faltung ist ebenfalls eine 3x3-Faltung statt der im Original verwendeten 2x2-Faltung.

5 Experimente

Es wurden drei Experimente mit dem vorgestellten Netz und den gegebenen Prostatadaten durchgeführt.

In Experiment 1 wurde untersucht, wie gut das auf Lungensegmentierung trainierte Netz Prostatabilder segmentieren kann. Da es schon vorher sicher war, dass dieses keine guten Ergebnisse liefern wird, wird hierrauf nicht näher eingegangen.

In Experiment 2 wird das vortrainierte Netz mit den Prostatabildern weiter trainiert und die Güte dieses neu trainierten Modells untersucht.

In Experiment 3 wird ein neues U-Net mit den Prostatabildern trainiert. Einige Layer werden jedoch mit den Wichtungen des trainierten Lungennetzes initialisiert.

Sämtliche Modelle werden mit den Einstellungen des GitHub-Projektes trainiert. Diese beinhalten die Architektur des Netzes sowie Optimierungsalgorithmus und Fehlerfunktion.

5.1 Experiment 2

5.1.1 Kreuzvalidierung

Um die Güte des zu trainierenden Modells vorher abzuschätzen, wird eine 10-fache Kreuzvalidierung auf der Trainingsmenge durchgeführt. Abbildung 5.1 zeigt die Ergebnisse der einzelnen Iterationen. Zeile 11 zeigt den zu erwartenden Generalisierungsfehler inklusive Standardabweichung. Dieser zeigt, dass das Modell ziemlich schlechte Ergebnisse mit sehr großen Schwankungen liefern wird, wenn es in dieser Form trainiert wird.

Row No.	CV-Errors
1	74.79814588193621
2	30.683067464894865
3	74.8200203355214
4	89.55307117218081
5	40.20664776469436
6	83.44051219451006
7	1.2246533212428634
8	48.05022818931739
9	0.8150463755005943
10	79.34992885803521
11	52.29413215578337 (+/- 31.61)

Abbildung 5.1: Fehler der Kreuzvalidierung, Generalisierungsfehler + Standardabweichung

5.1.2 Ergebnis

Abbildung 5.2 zeigt Beispieldaten des vollständig trainierten Netzes. Es sind sehr gute, aber auch nur mittelmäßige bis schlechte Ergebnisse zu erkennen.

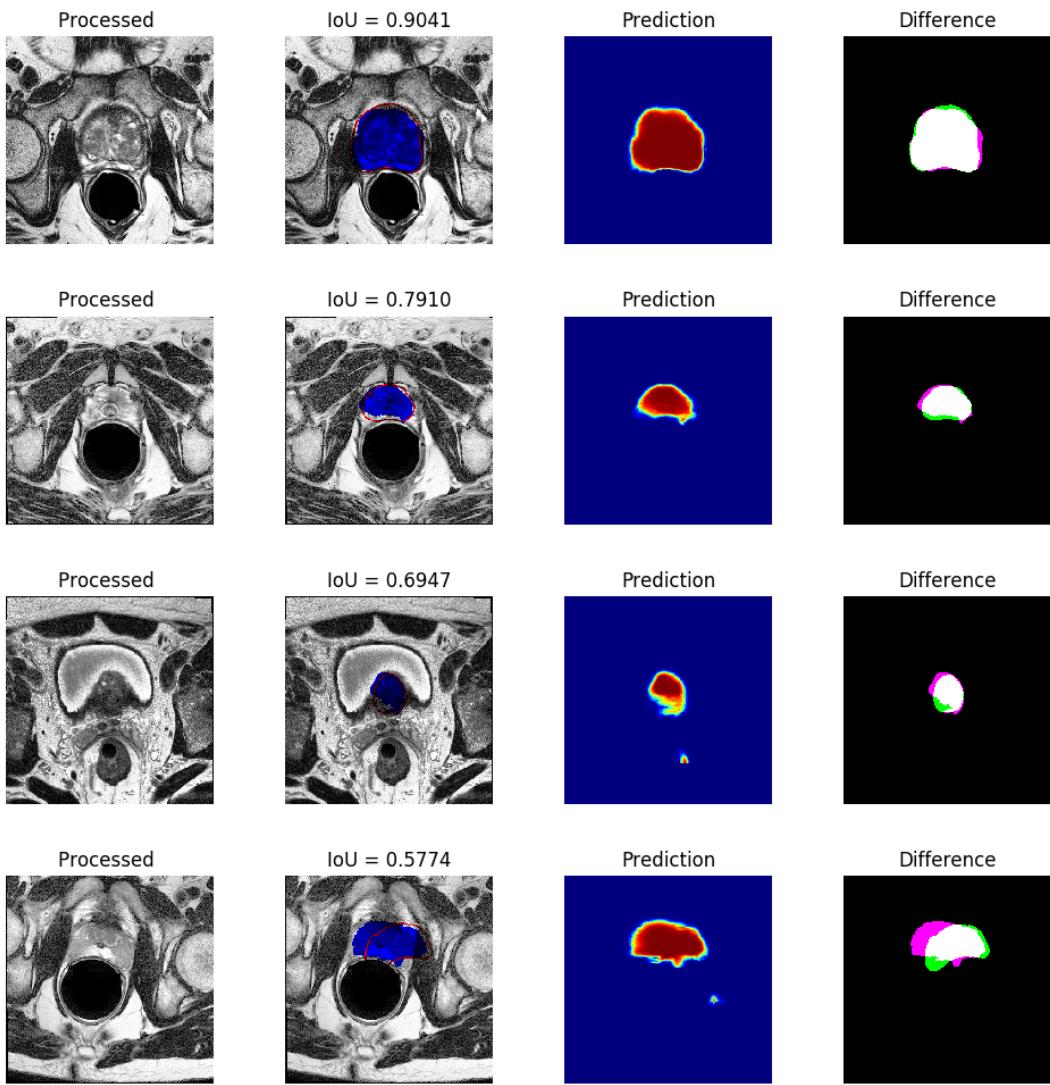


Abbildung 5.2: Beispielergebnisse Exp. 1

Abbildung 5.3 zeigt die Verteilung der IoU und Dice-Koeffizienten der Testmenge. Diese zeigt, dass einige Testpaare überhaupt nicht klassifiziert werden konnten. Dies könnte zusätzlich zu den vorhandenen Schwankungen der Ergebnisse darauf hinweisen, dass nicht genug Trainingsdaten vorhanden sind und der mehr Datenaugmentierungen notwendig sind.



Abbildung 5.3: IoU + Dice-Koeffizienten der Testmenge

5.2 Experiment 3

5.2.1 Wichtungsvergleich

Um herauszufinden, welche Wichtungen des Lungen-Netzes wiederverwendet werden können, werden die Wichtungen jedes Layers des Lungen-Netzes mit denen des Prostata-Netzes verglichen. Dazu wird der mittlere absolute Fehler der Wichtungsmatrizen jedes Layers berechnet. Dieser stellt in diesem Fall keinen Fehler dar, sondern den mittlere absoluten Abstand jeder Wichtung der Wichtungsmatrizen. In Abbildung 5.4 ist erkennbar, dass die Layer 10-13 sowie Layer 1 und 22 die geringsten Abstände haben im Verhältnis zu den restlichen Layern. Diese Layer haben sich beim Umtrainieren des Lungen-Netzes am wenigsten geändert. Somit werden diese Wichtungsmatrizen dieser Layer in das neue Netz übernommen.

Row No.	Layer	MAE
1	conv2d_1	0.010
2	conv2d_2	0.023
3	conv2d_3	0.026
4	conv2d_4	0.032
5	conv2d_5	0.032
6	conv2d_6	0.038
7	conv2d_7	0.041
8	conv2d_8	0.022
9	conv2d_9	0.016
10	conv2d_10	0.008
11	conv2d_11	0.008
12	conv2d_12	0.009
13	conv2d_13	0.011
14	conv2d_14	0.019
15	conv2d_15	0.024
16	conv2d_16	0.022
17	conv2d_17	0.024
18	conv2d_18	0.025
19	conv2d_19	0.020
20	conv2d_20	0.023
21	conv2d_21	0.024
22	conv2d_22	0.011

Abbildung 5.4: Wichtungsvergleich pro Layer des Lungen-Netzes mit Prostata-Netz

Da in diesem Versuch nur einige Wichtungsmatrizen übernommen werden, ist zu erwarten, dass die Ergebnisse dieses Netzes schlechter ausfallen als die des vorherigen Versuchs.

5.2.2 Ergebnis

Abbildung 5.5 zeigt Beispielergebnisse des trainierten Netzes. Wie auch in Experiment 2 sind gute, aber auch schlechte Ergebnisse zu erkennen.

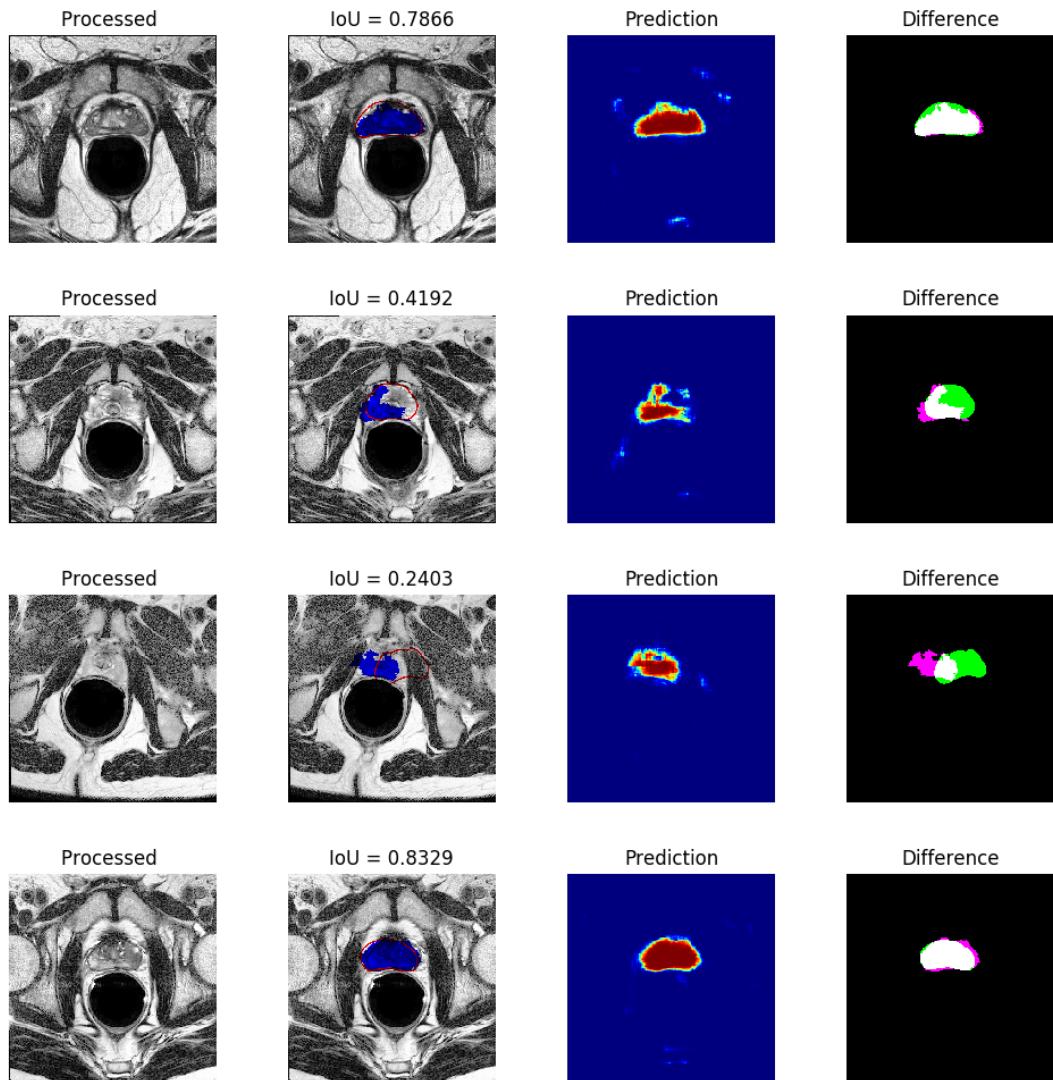


Abbildung 5.5: Beispielergebnisse Exp. 2

Auch die Verteilung der IoU und Dice-Koeffizienten entspricht prinzipiell der Verteilung aus Experiment 2 (Abbildung 5.6). Wie auch in diesem konnten einige Testpaare überhaupt nicht, andere hingegen sehr gut klassifiziert werden. Mittelwert und Standardabweichung des Gütemaße sind wie erwartet schlechter als in Experiment 2. Die liegt vermutlich an einer ungenügenden Menge der Trainingsdaten.



Abbildung 5.6: IoU + Dice-Koeffizienten Exp. 3

Vergleicht man erneut die Wichtungen der Netze, stellt man fest, dass sich die Wichtungen der vorher identifizierten Layer in diesem Layer nur wenig (Layer 1 und 22) beziehungsweise garnicht (Layer 10-13) geändert haben (Abbildung 5.7). Die bestätigt die Annahme, dass bestimmte Layer eines trainierten Deep-Learning-Netzes zum Training eines Neuen verwendet werden können, solange sie ähnliche Aufgaben erfüllen sollen.

Row No.	Layer	MAE
1	conv2d_1	0.017
2	conv2d_2	0.083
3	conv2d_3	0.081
4	conv2d_4	0.079
5	conv2d_5	0.076
6	conv2d_6	0.075
7	conv2d_7	0.052
8	conv2d_8	0.045
9	conv2d_9	0.040
10	conv2d_10	0.001
11	conv2d_11	0.000
12	conv2d_12	0.000
13	conv2d_13	0.000
14	conv2d_14	0.039
15	conv2d_15	0.038
16	conv2d_16	0.043
17	conv2d_17	0.051
18	conv2d_18	0.050
19	conv2d_19	0.054
20	conv2d_20	0.061
21	conv2d_21	0.066
22	conv2d_22	0.012

Abbildung 5.7: Wichtungsvergleich pro Layer des Lungen-Netzes mit Some-Layer-Prostata-Netz

6 Zusammenfassung

In diesem Projekt wurde ein DeepLearning neuronales Netz zur Segmentierung von Lungenflügeln in Brustkorb-Radiologieaufnahmen untersucht. Die Struktur dieses Netzes ist an das U-Net angelehnt. Das U-Net ist ein tiefes Convolutional neuronales Netz, das im Aufbau einem Autoencoder gleicht. Der Encoder-Teil ist über Konkatenationen mit dem entsprechenden Decoder-Teil der selben Ebene verbunden. Das vorgegebene Lungen-Netz wurde mit Prostata-Radiologieaufnahmen und entsprechenden Binärmasken weitertrainiert und die Güte dieses trainierten Netzes wurde gemessen. Beim Vergleichen der Wichtungen der einzelnen Layer des Netzes wurde festgestellt, dass Wichtungen bestimmte Layer des U-Nets beim Übertrainieren kaum geändert werden. Die Wichtungen dieser Layer wurden in einem zweiten Versuch in ein neues U-Net übernommen, das erneut auf Segmentierung der Prostata trainiert wurde. Ein anschließender Wichtungsvergleich zeigte, dass die übernommenen Layer wieder kaum Änderungen erfuhren.

Literatur

- [Pet17] Vitali Petsiuk. *Lung Segmentation (2D)*. 2017.
- [RFB15] Olaf Ronneberger, Philipp Fischer und Thomas Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: *CoRR* abs/1505.04597 (2015). arXiv: 1505.04597. URL: <http://arxiv.org/abs/1505.04597>.
- [Ron+15] Olaf Ronneberger u. a. *Our U-net wins two Challenges at ISBI 2015*. 2015.

Abbildungsverzeichnis

2.1	Beispiel - Prostataaufnahme	2
2.2	Beispiel - Binärmaske	3
3.1	U-Net Architektur [RFB15]	4
4.1	Keras-Modell [Pet17]	7
5.1	Fehler der Kreuzvalidierung, Generalisierungsfehler + Standardabweichung	10
5.2	Beispielergebnisse Exp. 1	11
5.3	IoU + Dice-Koeffizienten der Testmenge	11
5.4	Wichtungsvergleich pro Layer des Lungen-Netzes mit Prostata-Netz	12
5.5	Beispielergebnisse Exp. 2	13
5.6	IoU + Dice-Koeffizienten Exp. 3	14
5.7	Wichtungsvergleich pro Layer des Lungen-Netzes mit Some-Layer-Prostata-Netz	14