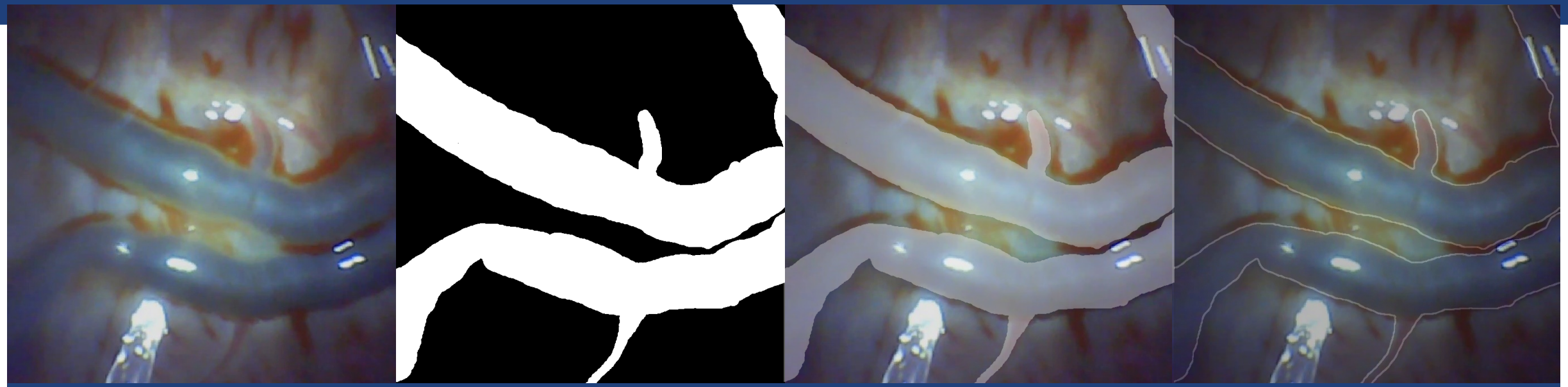
ETH zürich



Automated vessel detection for fetal surgery

Jonas Bohn

Bachelor Thesis, Final presentation

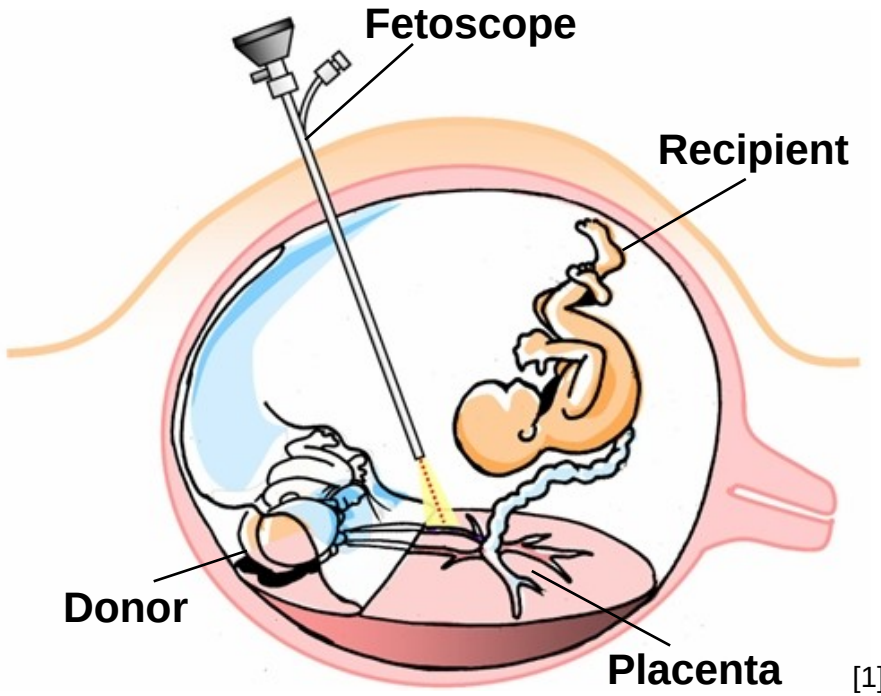
Multi-Scale Robotics Lab, ETH Zurich

11. Februar 2025

Supervisors: Jonas Lussi, Simone Gervasoni

Medical Background

- Twin to Twin Transfusion Syndrome (TTTS)
- 10 to 15% of monochorionic twins affected
- Imbalance of blood flow
- Risk of death of both twins
- Fetoscopic laser coagulation

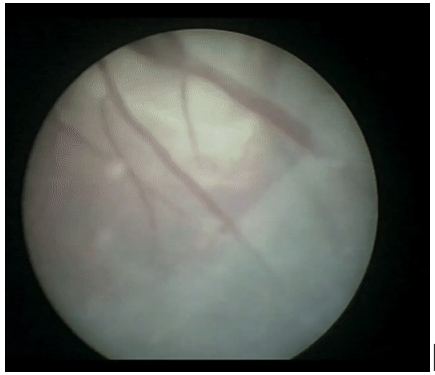


[1]

[1] F. I. Luks. Schematic illustration of endoscopic fetal surgery for twin-to-twin transfusion syndrome, 2009.

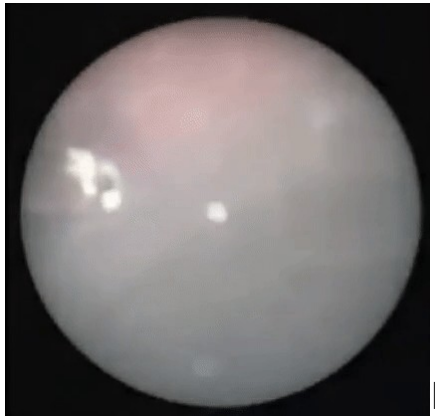
Fetoscopic Laser Coagulation

- Visually explore placenta with camera
- Revisit anastomoses and ablate them with laser


[2]

Challenges:

- Limited field-of-view
- Occlusions
- Poor visibility



[3]

[2] S.Fleury, https://www.youtube.com/watch?v=VLdZAZck2dU, [online accessed: 22.01.2021]
[3] S.Bano, Deep placental vessel segmentation for fetoscopic mosaicking, MICCAI2020

Motivation

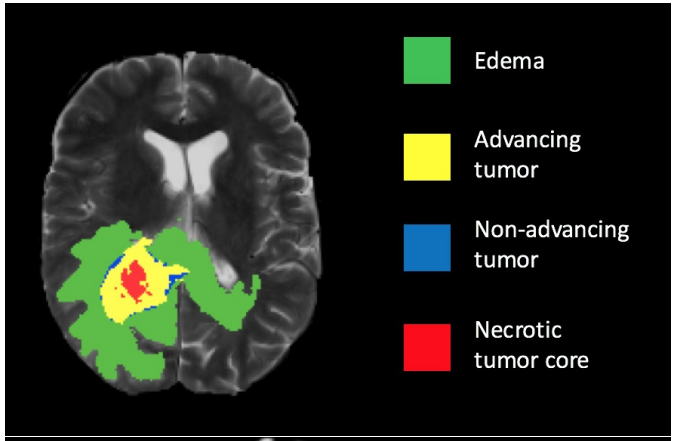
- Robust detection of blood vessels
- Enhance the surgeon view
- Minimize complications after surgery
- Segmentation

[4]

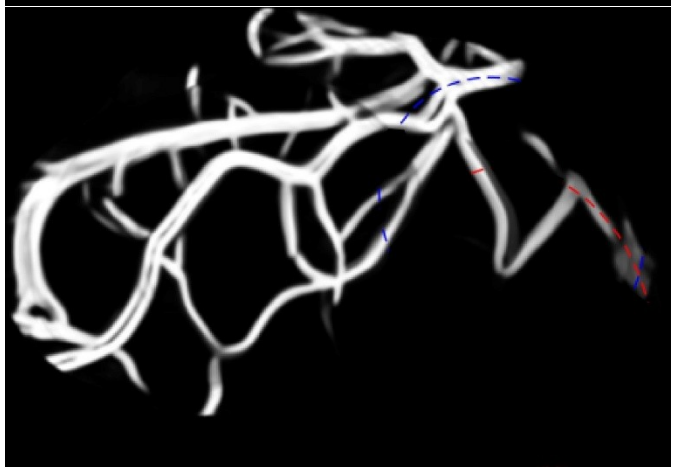
[4] S.Bano, Deep placental vessel segmentation for fetoscopic mosaicking, MICCAI2020

State of the art

- Deep Learning outperforms other
- Convolutional Networks
- Segmented placental vessel maps
- Limited data



[5]



[6]

[5] M.Havaei, Brain Tumor Segmentation with Deep Neural Networks, 2016
[6] S.Bano, Deep placental vessel segmentation for fetoscopic mosaicking, MICCAI2020

Datasets

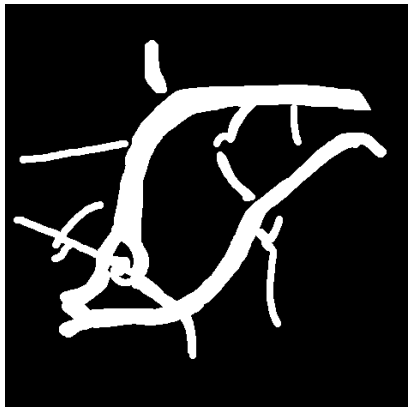
UCL dataset:

- 482 video frames
- 6 in vivo procedures
- Verified by expert

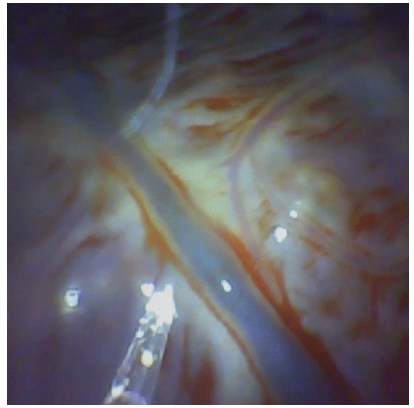
MSRL dataset:

- 269 video frames
- Ex vivo

Input image | Ground truth

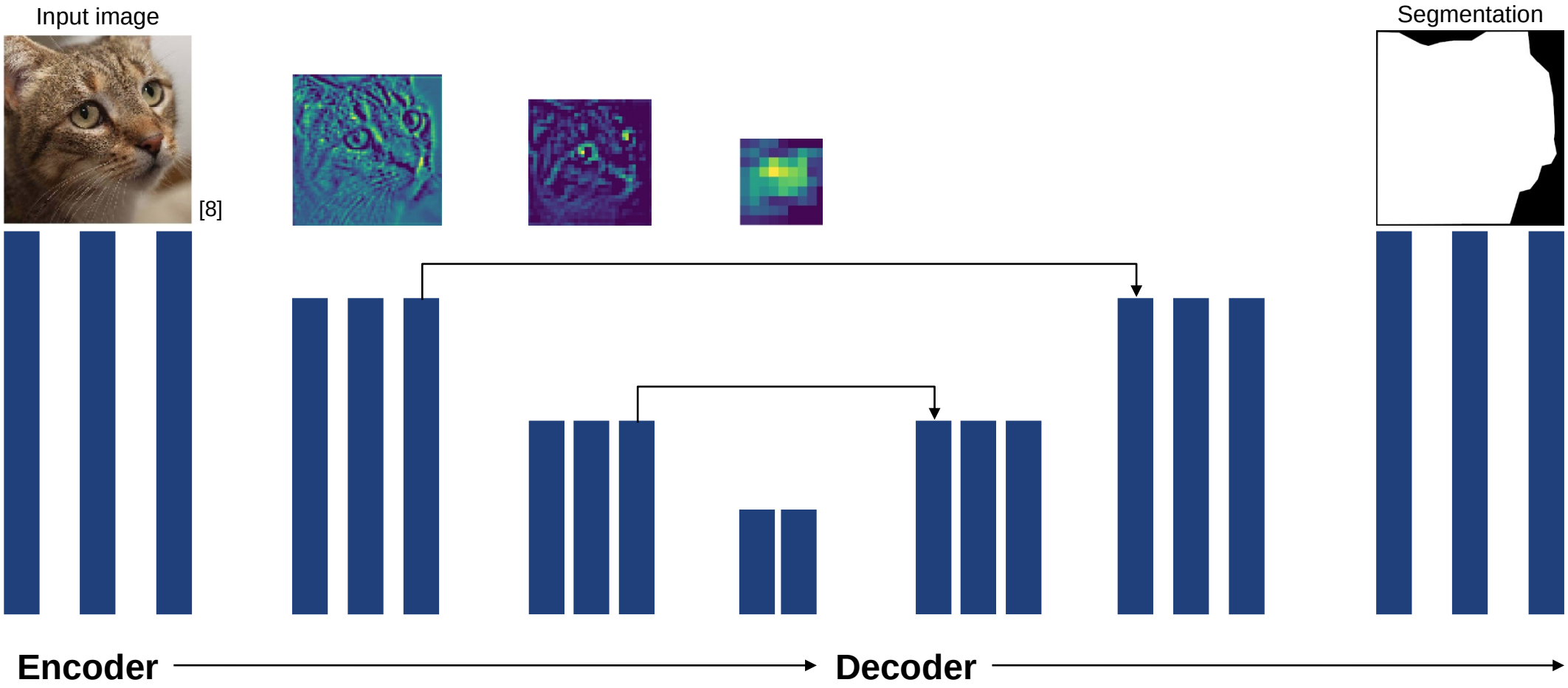


[7]



[7] S.Bano, Deep placental vessel segmentation for fetoscopic mosaicking, MICCAI2020

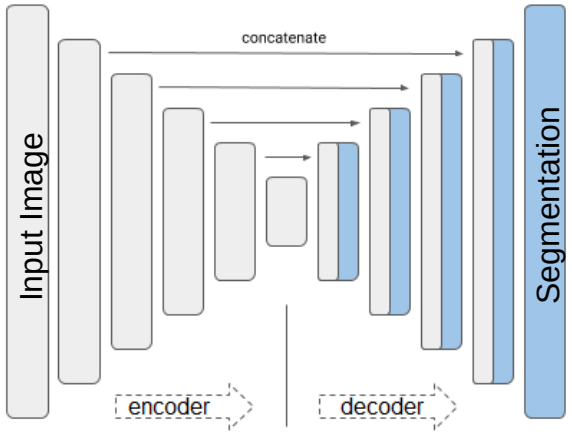
Encoder-Decoder Network



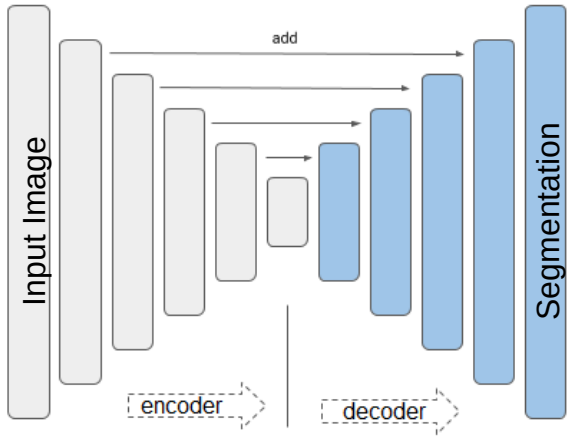
[8] A.Dertat, Applied Deep Learning - Part 4: Convolutional Neural Networks, 2017

Architectures

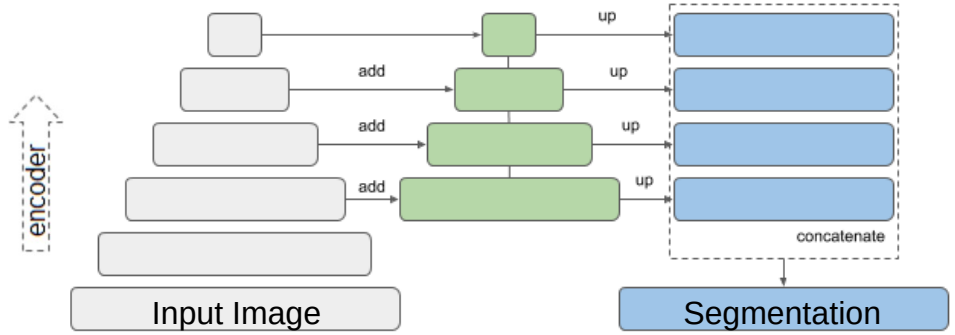
- U-Net



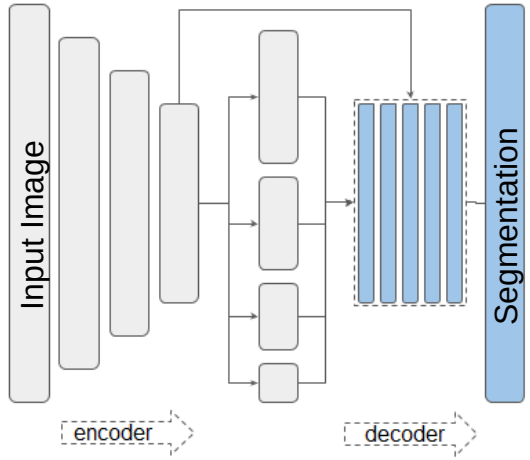
- LinkNet



- FPN



- PSPNet



[9]

[9] P. Yakubovskiy. Segmentation models. `https://github.com/qubvel/segmentation_models`, 2019

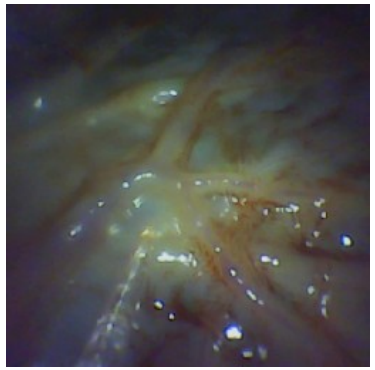
Experimental Setup

- 6-fold crossvalidation
- 26 networks
- Data augmentation

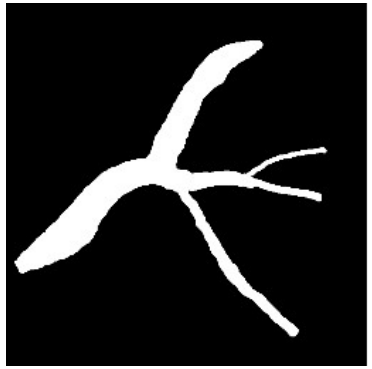
U-Net		Fold 1
No. Train images		1526
No. validation images		52
Vanilla	Dice	0.69 (+/- 0.13)
	IoU	0.55 (+/- 0.15)
VGG16	Dice	0.70 (+/- 0.12)
	IoU	0.56 (+/- 0.14)
DenseNet201	Dice	0.58 (+/- 0.22)
	IoU	0.44 (+/- 0.21)
EfficientNetB5	Dice	0.67 (+/- 0.19)
	IoU	0.53 (+/- 0.20)
InceptionResNetV2	Dice	0.68 (+/- 0.17)
	IoU	0.53 (+/- 0.18)
InceptionV3	Dice	**0.73 (+/- 0.13)**
	IoU	**0.59 (+/- 0.16)**
ResNet101	Dice	0.58 (+/- 0.23)
	IoU	0.45 (+/- 0.23)

Segmentation Results

Input Image



Ground truth



Prediction

Baseline

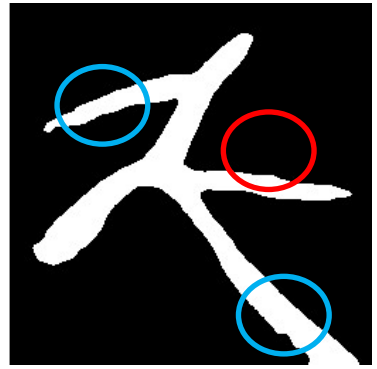
UNet-EffNetB5

PSP-InceptionV3

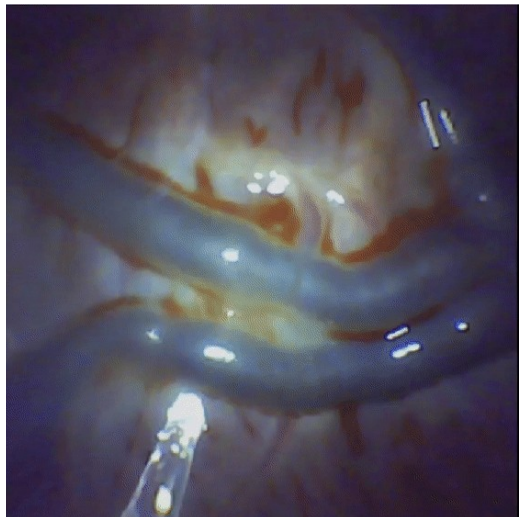
LinkNet-EffNetB5

FPN-DenseNet201

FPN-EffNetB5

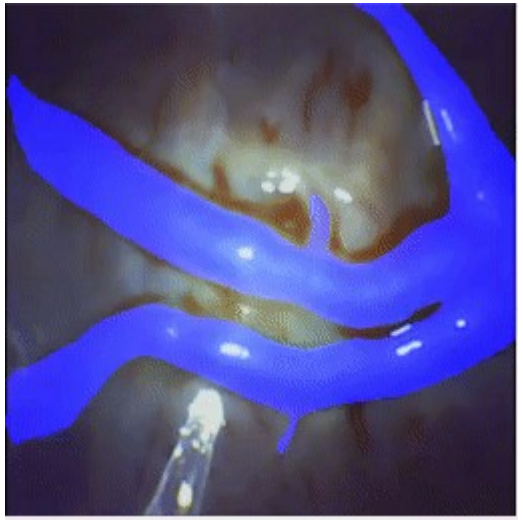


Real time segmentation

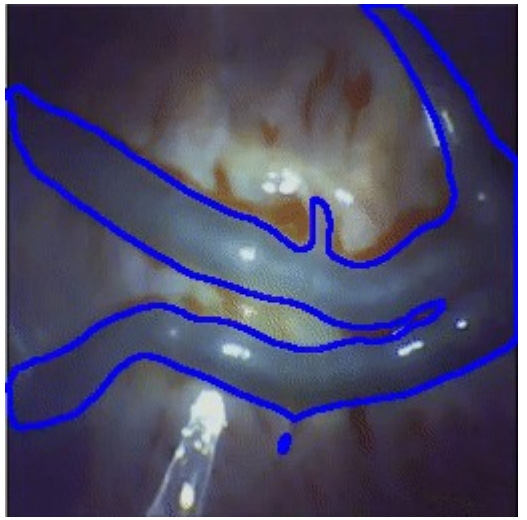


Original video

Predicted segmentation



Overlay

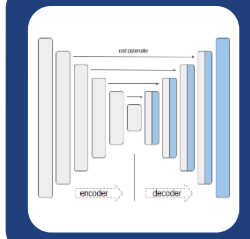


Contour

Conclusion & Future Work

New dataset

Increase data quality



Compared networks

Integrate network to system

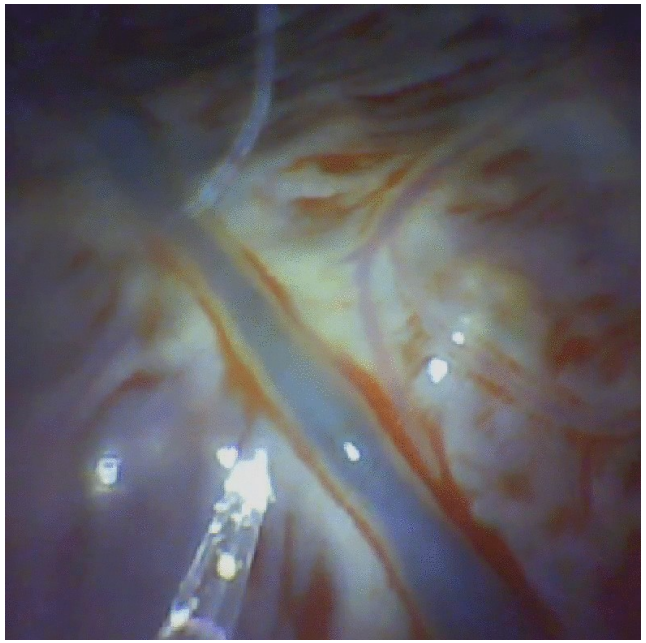
Automated detection

Vessel maps

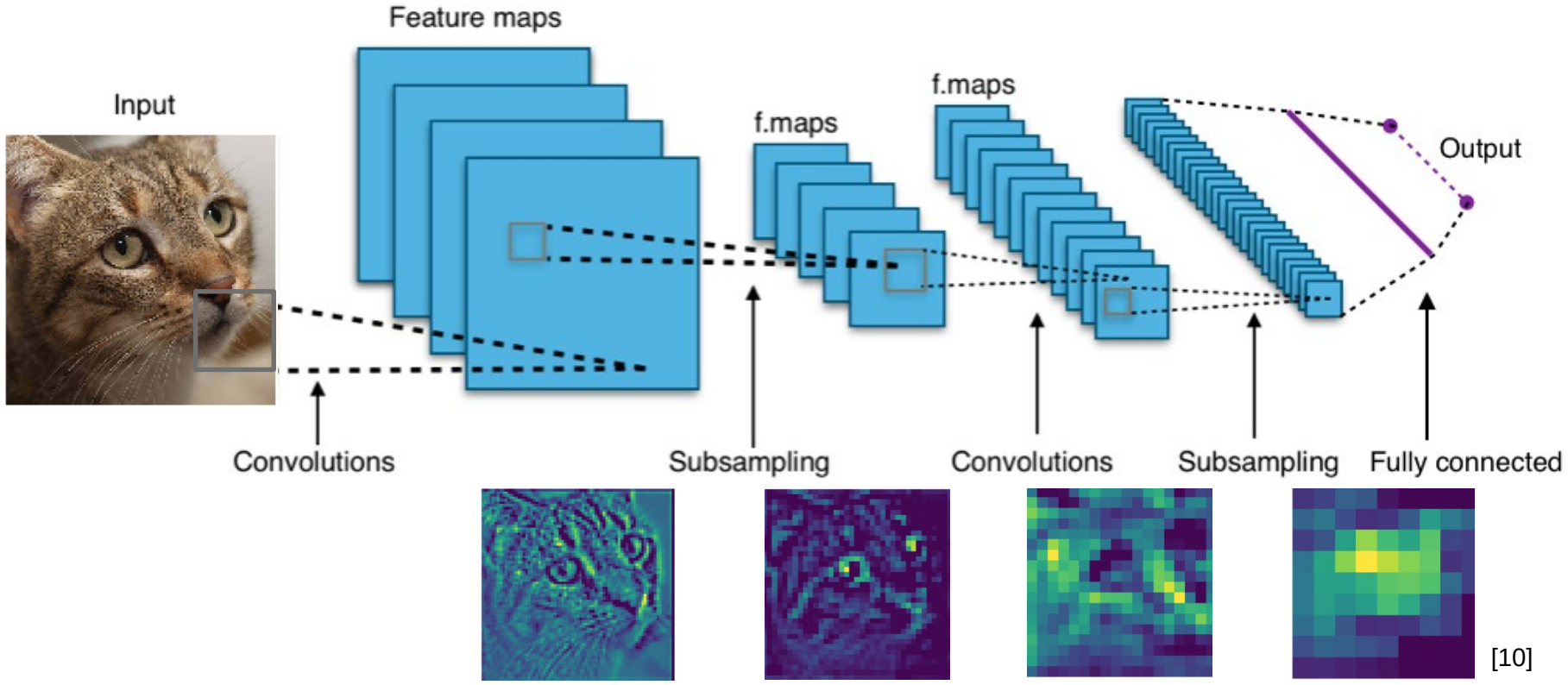
Acknowledgements

- Prof. Dr. Bradley Nelson
- Jonas Lussi
- Simone Gervasoni

Thanks for listening



Further explanation CNN



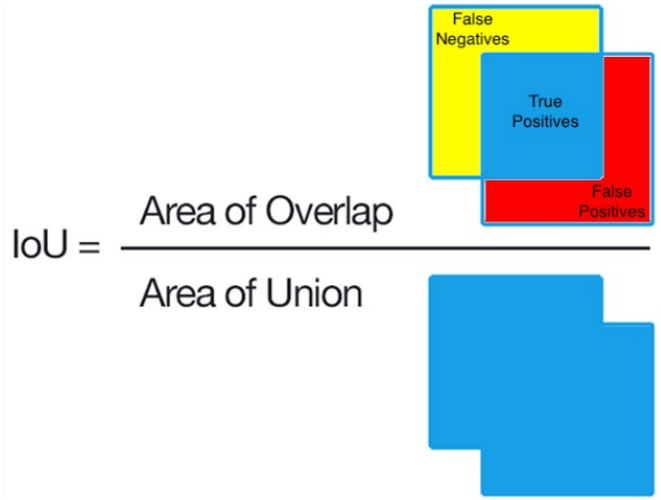
[10]

More information on image → More information on class

[10] S. Dertat, Applied Deep Learning - Part 4: Convolutional Neural Networks, 2017

Further explanation Crossvalidation

U-Net		Fold 1	...	Fold 5	Fold 6	Overall
No. Train images		1526		1631	1575	
No. validation images		52		36	44	270
Vanilla	Dice	0.69 (+/- 0.13)		0.58 (+/- 0.20)	0.67 (+/- 0.16)	0.70 (+/- 0.17)
	IoU	0.55 (+/- 0.15)		0.44 (+/- 0.19)	0.53 (+/- 0.18)	0.58 (+/- 0.18)
VGG16	Dice	0.70 (+/- 0.12)		0.57 (+/- 0.20)	0.67 (+/- 0.18)	0.70 (+/- 0.17)
	IoU	0.56 (+/- 0.14)		0.42 (+/- 0.19)	0.53 (+/- 0.18)	0.58 (+/- 0.18)
DenseNet201	Dice	0.58 (+/- 0.22)		0.58 (+/- 0.26)	**0.73 (+/- 0.12)**	0.71 (+/- 0.18)
	IoU	0.44 (+/- 0.21)		0.46 (+/- 0.25)	**0.59 (+/- 0.14)**	0.58 (+/- 0.19)
EfficientNetB5	Dice	0.67 (+/- 0.19)		**0.66 (+/- 0.16)**	0.73 (+/- 0.14)	**0.73 (+/- 0.15)**
	IoU	0.53 (+/- 0.20)		**0.51 (+/- 0.17)**	0.59 (+/- 0.16)	**0.61 (+/- 0.17)**
InceptionResNetV2	Dice	0.68 (+/- 0.17)		0.56 (+/- 0.23)	0.70 (+/- 0.13)	0.70 (+/- 0.17)
	IoU	0.53 (+/- 0.18)		0.42 (+/- 0.21)	0.55 (+/- 0.15)	0.58 (+/- 0.18)
InceptionV3	Dice	**0.73 (+/- 0.13)**		0.59 (+/- 0.22)	0.72 (+/- 0.13)	0.73 (+/- 0.16)
	IoU	**0.59 (+/- 0.16)**		0.45 (+/- 0.20)	0.58 (+/- 0.15)	0.61 (+/- 0.17)
ResNet101	Dice	0.58 (+/- 0.23)		0.55 (+/- 0.23)	0.70 (+/- 0.16)	0.69 (+/- 0.19)
	IoU	0.45 (+/- 0.23)		0.42 (+/- 0.20)	0.55 (+/- 0.18)	0.57 (+/- 0.20)



$$Dice = \frac{2 \times TP}{(TP + FP) + (TP + FN)}$$