Catheter segmentation in X-ray fluoroscopy using synthetic data and transfer learning with light U-nets

Marta Cherardini, Evangelos Mazomenos, Arianna Menciassi, Danail Stoyanov

Seminar Presentation

Kinjal Shah

April 16, 2020

Team 9

Project Summary: Detection and Guidance of K-Wire Placement in Pelvic Trauma Surgery

Team 9: Kinjal Shah, Irina Bataeva
Mentor: Dr. Ali Uneri

Motivation: Improve accuracy to reduce injury while limiting radiation exposure

Objective: Use emerging deep learning methods to:

(1) Detect K-wires in 2D radiographs of the pelvis

(2) Localize their 3D pose to provide surgical navigation during fracture fixation



Background and Paper Selection

- Fluoroscopy is a widely used imaging modality for medical procedures
- Deep learning has enabled highly accurate image segmentation modalities
- U-Net architecture developed, providing high accuracy image segmentation for medical images in particular
- In current processes there are trade offs between accuracy and speed

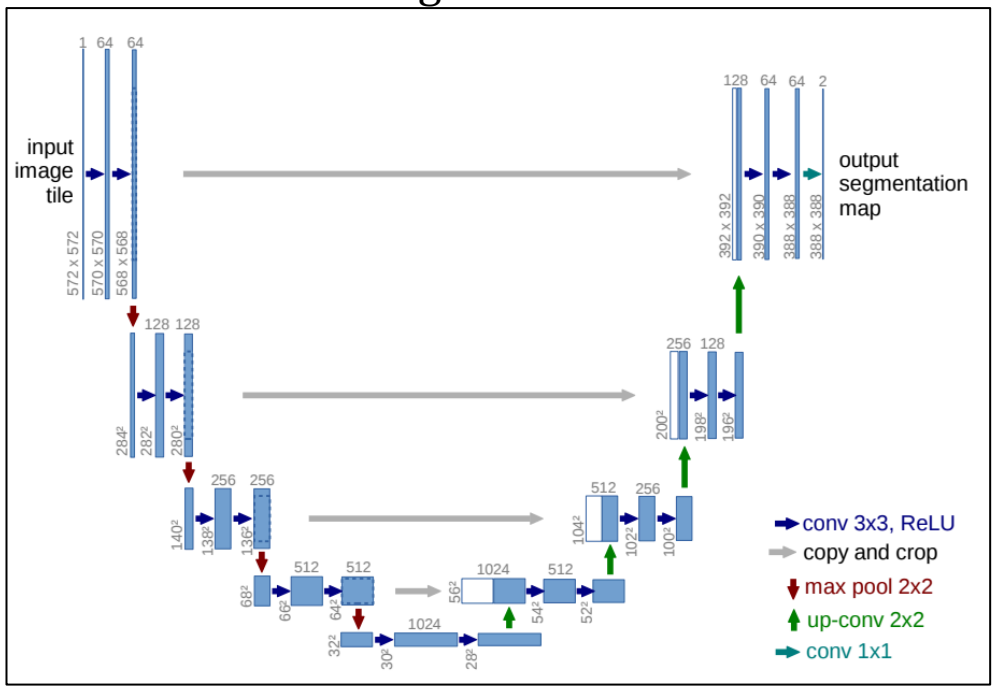
Catheter segmentation in X-ray fluoroscopy using synthetic data and transfer learning with light U-nets

2020 Computer Methods and Programs in Biomedicine

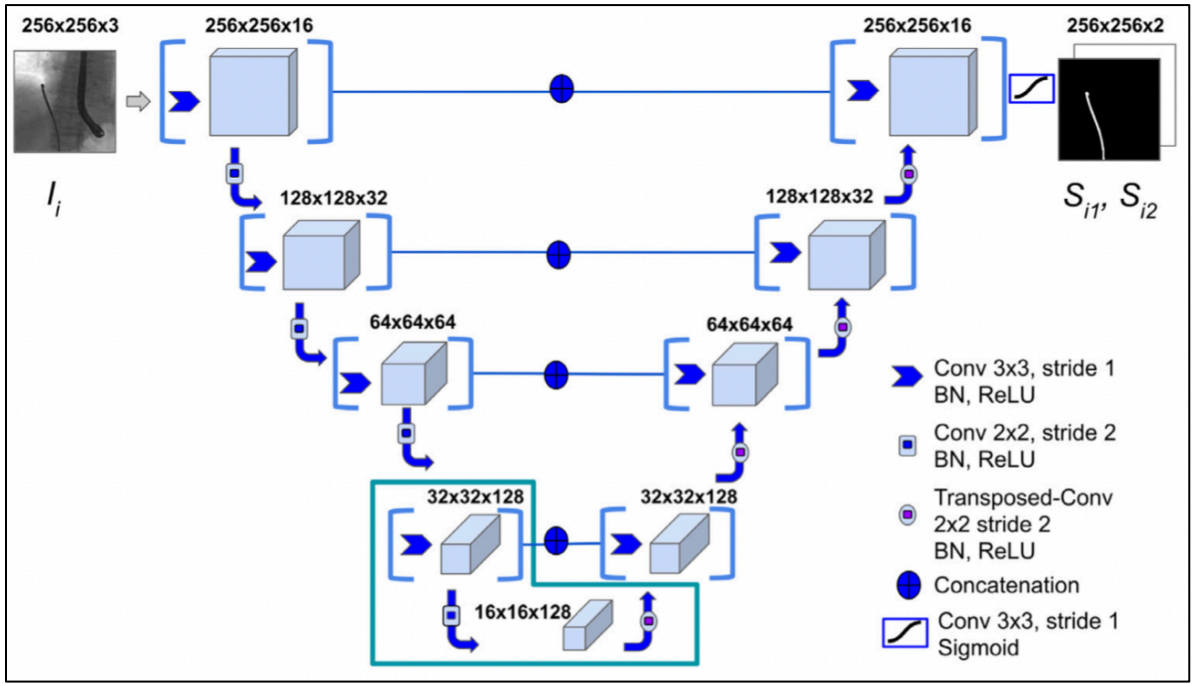
- Lightweight U-Net architecture
- Synthetic Dataset for Training
- Fine-tuning deepest layers using images similar to target test set
- Transfer Learning

Method: CNN Model

Original UNet[2]



Lightweight UNet[1]



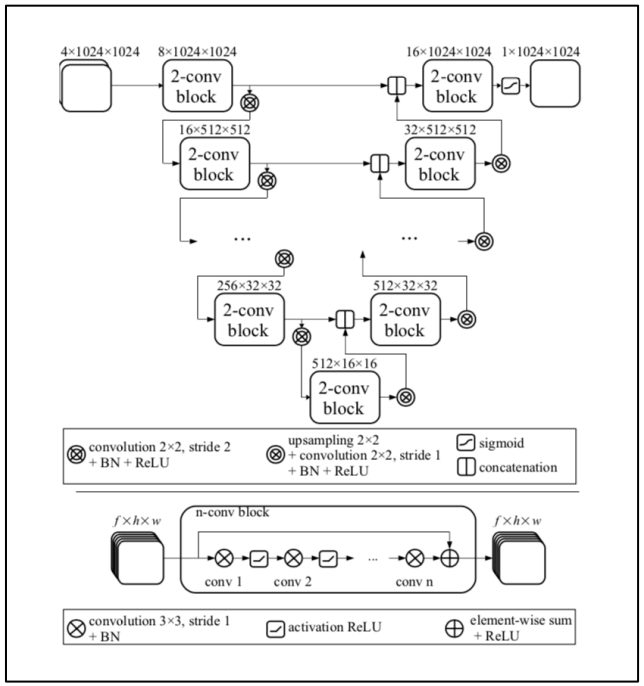
- Reduced convolutions per layer from 2 to 1
- Convolutional layers followed by batch normalization
- ReLU activation function
- Last layer: sigmoid activation function applied to provide pixel-wise classification

[1] Gherardini, Marta, et al. “Catheter Segmentation in X-Ray Fluoroscopy Using Synthetic Data and Transfer Learning with Light U-Nets.” Computer Methods and Programs in Biomedicine, vol. 192, 2020, p. 105420., doi:10.1016/j.cmpb.2020.105420.

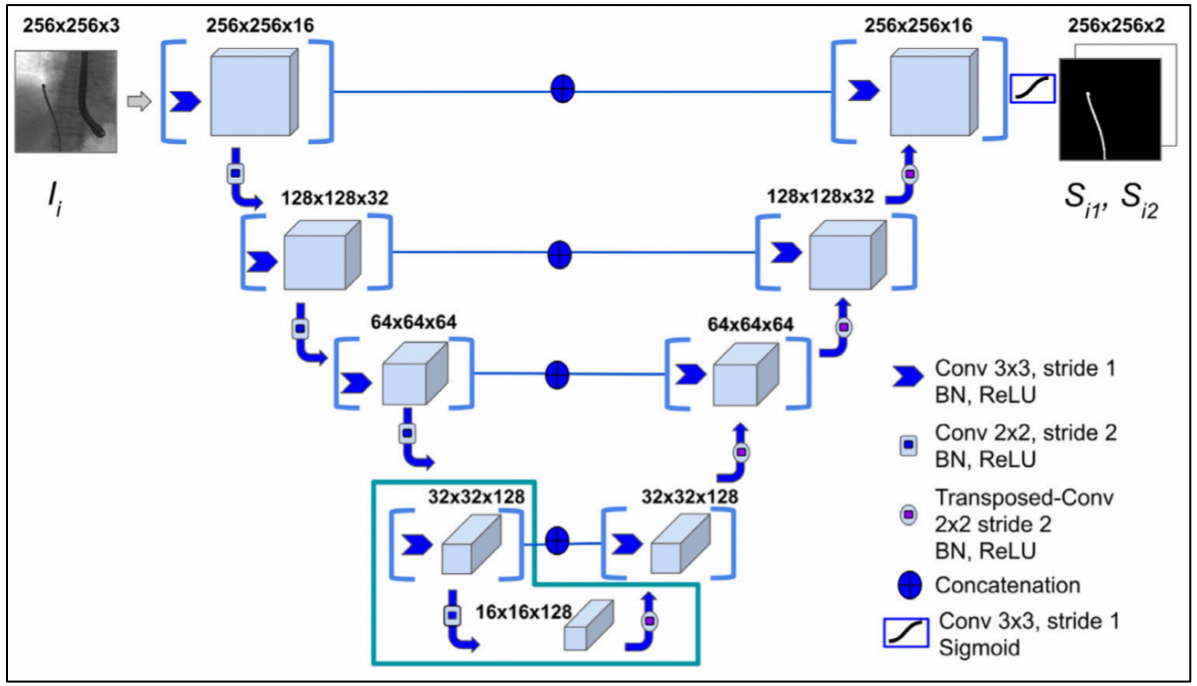
[2] Ronneberger, Olaf, et al. “U-Net: Convolutional Networks for Biomedical Image Segmentation.” *Lecture Notes in Computer Science Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, 2015, pp. 234–241., doi:10.1007/978-3-319-24574-4_28.

Method: CNN Model

Ambrosini et al.[2]



Lightweight UNet[1]



- Reduced convolutions per layer from 2 to 1
- Reduced layers from 110 to 55
- Reduced pixel size from 1024x1024 to 256x256

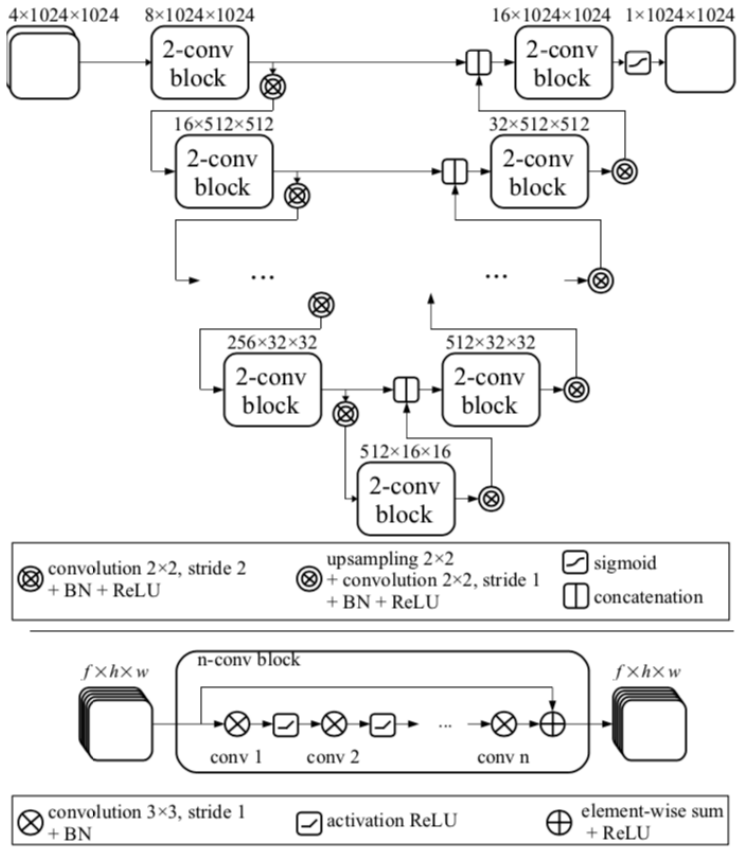
[1] Gherardini, Marta, et al. “Catheter Segmentation in X-Ray Fluoroscopy Using Synthetic Data and Transfer Learning with Light U-Nets.” Computer Methods and Programs in Biomedicine, vol. 192, 2020, p. 105420., doi:10.1016/j.cmpb.2020.105420.
[2] P. Ambrosini, D. Ruijters, W. Niessen, A. Moelker, and T. Walsum, “Fully automatic and real-time catheter segmentation in x-ray fluoroscopy,” 09 2017, pp. 577–585.

Note: Quick Summary of a Referenced Paper

Fully automatic and real-time catheter segmentation in X-Ray fluoroscopy

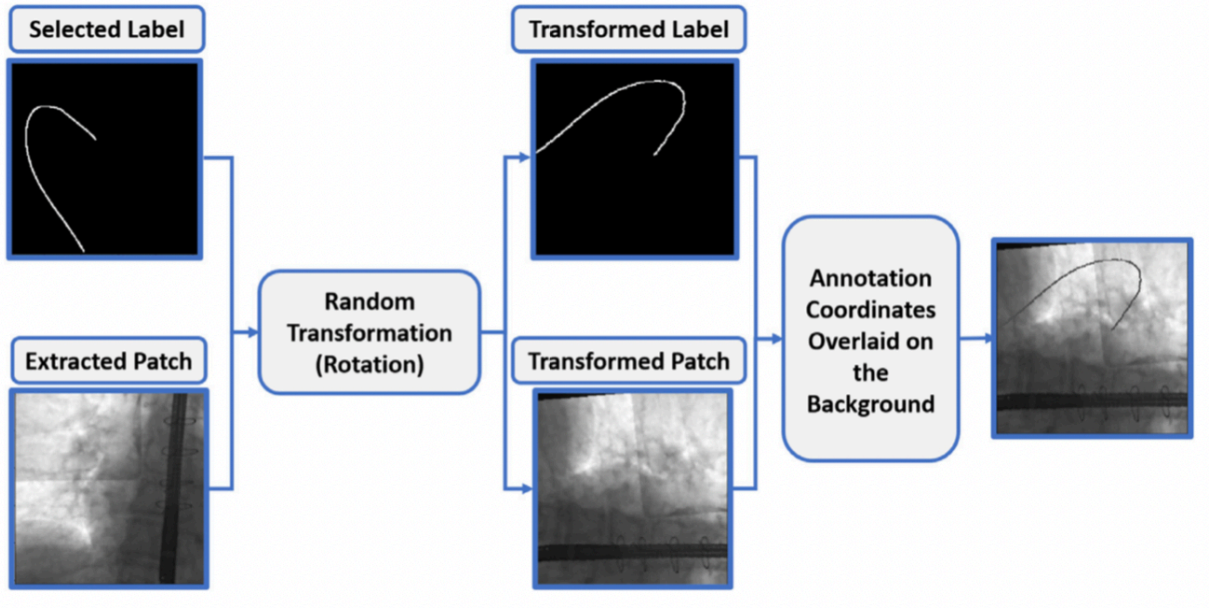
Pierre Ambrosini, Daniel Ruijters, Wiro J. Niessen, Adriaan Moelker, and Theo van Walsum

- Ambrosini et al. perform catheter and guidewire segmentation using Unet architecture
- Leverage temporal information from video (current image and 3 prior) as input into network
- Leverage small annotated dataset of real images along with data augmentation
- Light U-Net paper considers this paper the state-of-the-art for real-time surgical instrument segmentation
- Paper proves value by comparing metrics to this paper



Method: Dataset

Leveraged 3 different datasets for experiments



[1] Gherardini, Marta, et al.

- Dataset-1: 9000 Synthetic Images
  - Ground truth known
  - Patch extracted from non-catheter containing sections of dataset-3 images
  - Masks from Dataset-2 and 3 are selected
  - Masks and patch are paired and undergo random transformation
  - Coordinates of catheter are projected onto background image
- Dataset-2: 2000 Phantom Images
  - Segmented semi-automatically
  - Recorded from 4 procedures performed on silicone aorta phantoms
- Dataset-3: 1207 Real *in-vivo* images
  - Collected from 6 cardiovascular procedures (T1-T6)
  - T1 manually annotated
  - T2-T6 Semi-automatically annotated

Method: Training

1. End-to-End training on synthetic and phantom datasets (dataset-1, dataset-2)
   - Training images shuffled and normalized
2. Fine-tuning on deepest 7 layers with portion of *in-vivo* images (dataset-3)
   - Dataset-3 is divided into 3 split groups
     - S1 – 25% of image from each of 6 subsets T1-T6 for fine-tuning (240 images), rest used for testing
     - S2 – T1,T4, T6 kept for fine-tuning (714 images), rest for testing
     - S3 – T1,T2,T5 kept for fine-tuning (579 images), rest for resting

Experiments and Results

Experiment	Result	Takeaway
Two Experiments run simultaneously: 1. Trained on synthetic dataset 2. Trained on phantom dataset	Both datasets performed similarly	A large simulated dataset can perform well
Fine-tuning dataset selection: S1, S2, and S3 compared for impact	S1 (images selected from all subsets T1-T6)	The more similar the fine-tuning images are to the test set, the better the model performs
Leave One Out: Real images only (1207 images were split between train and test sets)	Model did not converge	Large simulated dataset performs better than small real dataset

Table 4
Segmentation accuracy on the leave-one-out (LOO) experiment and on the network fine-tuned on splits S2 and S3, for both *Experiment-1* and *Experiment-2*. Each row reports the Dice coefficient separately for each testing dataset, as well as the average Dice for each test.

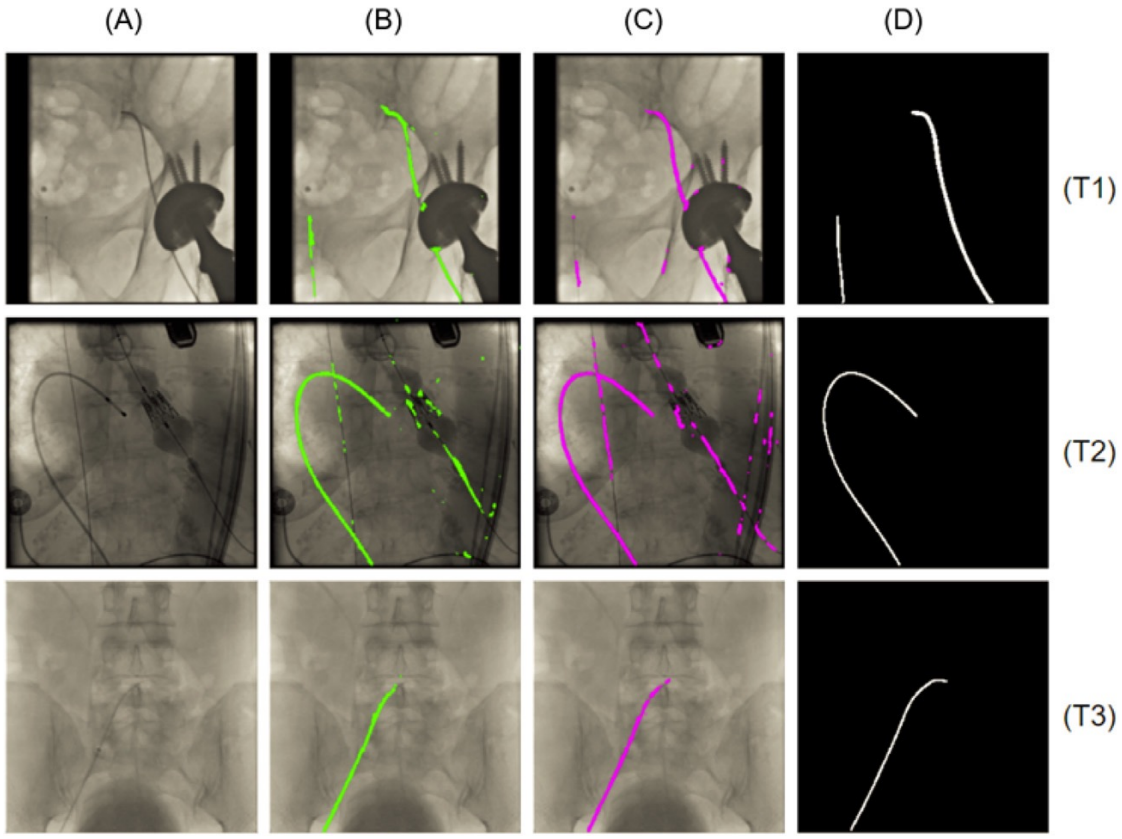
Test	T1	T2	T3	T4	T5	T6	Avg.
LOO	0.03	0.29	0.20	1×10^{-4}	1×10^{-4}	0.10	0.10
S1 - Exp. 1	-	0.31	0.31	-	0.16	-	0.26
S2 - Exp. 1	-	-	0.36	0.02	-	0.11	0.15
S1 - Exp. 2	-	0.28	0.33	-	0.15	-	0.25
S2 - Exp. 2	-	-	0.36	0.07	-	0.13	0.19

Table 5
Segmentation results for the network fine-tuned on S1. Dice coefficient and percentage of misclassified pixels in terms of False Positives (FPs) for *Experiment 1* (CNN trained on synthetic data) and *Experiment 2* (CNN trained on phantom data), after fine-tuning on S1.

Dataset	Dice Exp.1	Dice Exp.2	FPs (%) Exp.1	FPs (%) Exp.2
T1	0.58	0.57	0.98	1.18
T2	0.78	0.76	2.35	2.85
T3	0.72	0.77	1.08	1.14
T4	0.71	0.73	1.82	2.55
T5	0.29	0.29	1.48	1.97
T6	0.13	0.18	1.33	2.10

[1] Gherardini, Marta, et al.

Results



- A: Grayscale input image
- B: Train with synthetic images + fine tune
- C: Train with phantom images + fine tune
- D: Ground Truth Mask

[1] Gherardini, Marta, et al.

Comparison to Ambrosini et al.

Experimental Setup

- Implemented Ambrosini et al's model on their data set
- Performed Exp.1 and 2 (simulated training data and phantom training data)
- Fine-tuned the last 7 layers in a similar manner using the S1 fine-tune:test set split

Results

- Comparable Dice coefficients (4-5% difference for both training datasets)
- 84% reduction in testing time

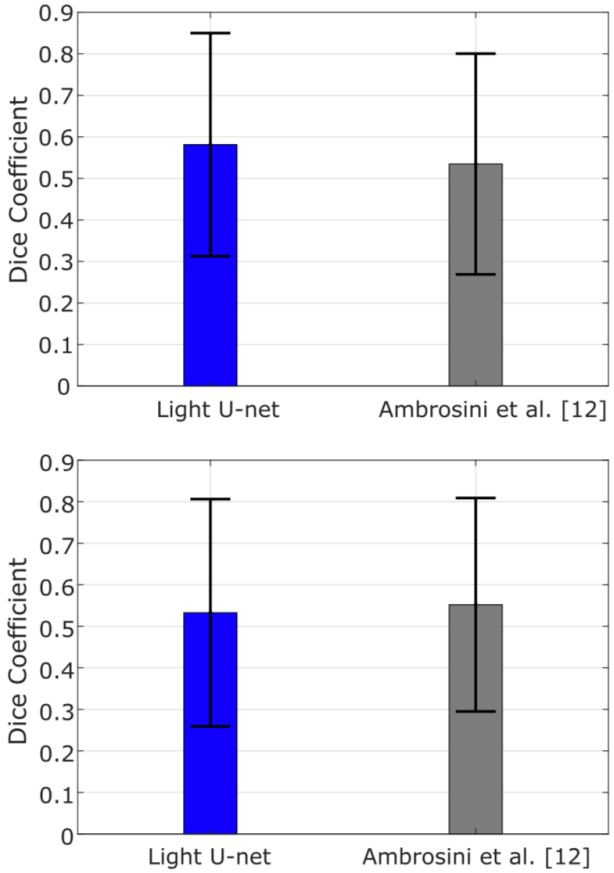


Fig. 7. Testing accuracy (Dice coefficient) between the proposed lightweight U-Net architecture and the one in [12] for *Experiment-1* (top) and *Experiment-2* (bottom).

Table 6
Number of trainable parameters and average testing time for the proposed lightweight U-Net and the one in [12].

CNN architecture	#Parameters	Average Time
Ambrosini et al. [12]	14,133,154	451 ± 12 ms
Light U-Net	687,634	71 ± 2 ms

[1] Gherardini, Marta, et al.

Review: Strengths and Takeaways

- Paper outlines and evaluated a semantic segmentation method using a synthetic test set
- Via parallel comparison to established state-of-the-art paper, **prove that there is an opportunity for achieving faster segmentation without a loss in accuracy**
- Provides baseline metrics for our evaluation: we will be attempting to not use any *in-vivo* images during training
- Lightweight architecture achieves goals of increased speed with comparable accuracy
- Very clear description of experimental setup and dataset generation

Review: Weaknesses

- Accuracy Measurement
  - Rely solely on Dice coefficients to compare with the work of Ambrosini et al.
  - Ambrosini et al. evaluated accuracy using **centerline and median tip distance**
- Selection of S1 in defining their proof of concept
  - Model was fine-tuned on images very similar (in some cases nearly identical) to the images it will be evaluating
  - Unclear if this increased performance accuracy as a test run on a completely new test set was not run

Lightweight model and transfer learning approach show promise but need to be more robustly evaluated

Questions?