# K2S: From Undersampled K-space to Automatic Segmentation

# MICCAI 2022

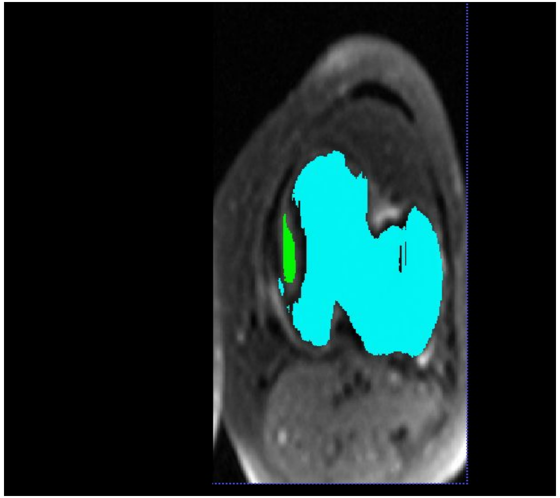
Quintin van Lohuizen & Stefan Fransen
PhD Candidates
Department of Radiology
q.y.van.lohuizen@umcg.nl & s.j.fransen@umcg.nl
**U**niversity **M**edical **C**enter **G**roningen (UMCG)

# Purpose

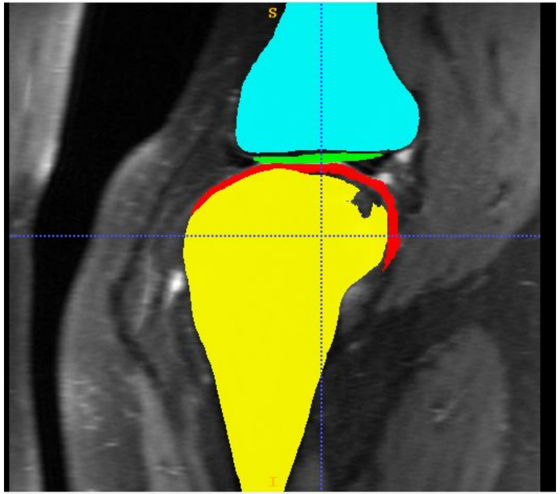
- FastMRI: Accelerating MRI for diagnosis and interventions of prostate cancer
  - Three medical institutes (UMCG, RUMC & TU)
  - Sharing: Data, Algorithms and Expertise
- Undersample in k-space
  - Reduced health-care costs
  - Less time in the machine
- Automatic segmentation
  - Lack of standardization
  - Tedious manual post-processing

# Data

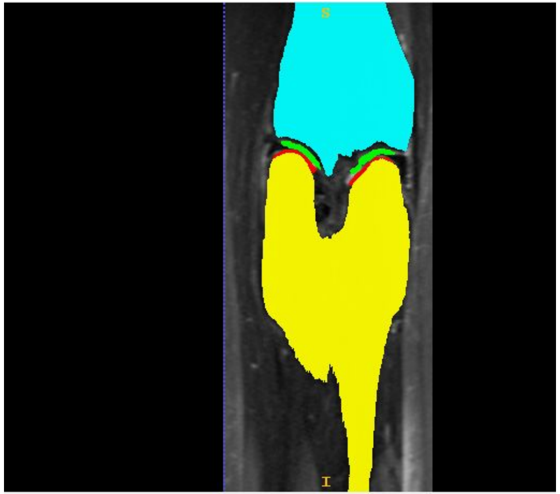
Transversal



Sagittal



Coronal



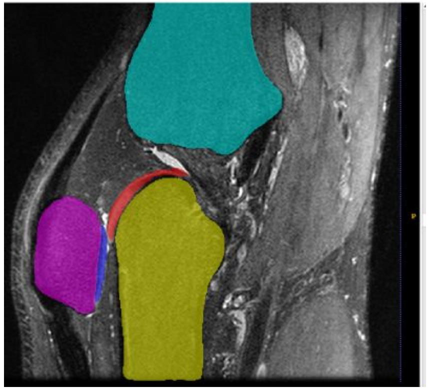
- 300 knee MRI k-spaces
  - K-space: 18-coils with acquisition matrices: (256, 256, 200)
- One k-space mask (8x)
- Model generated ground truth
  - The AI can only be as good as the data you give it

# — Approach

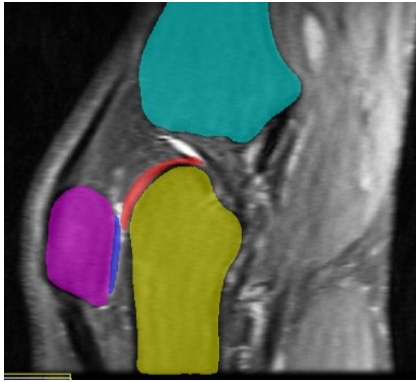
## **Pre-processing** - Deep Learning - Post-processing

- High dimensional k-space data
  - Patch-based training
    - Able to process varying input sizes
- Going to image domain
  - Zero padding in k-space to: (512, 512, 196)
  - Reconstruction: Root Sum of Squares (RSS)
    - Absence of coil sensitivity maps
- Normalization (division by 1000)

Normal



8x undersampled



# — Approach



- State-of-the-Art 3D U-Net (*Saha et al., 2021*)
  - Squeeze-and-excite attention layer (*Hu et al., 2019*)
  - Used for prostate cancer detection
- Weighted dice loss:

$$\mathcal{L}_{Dice} = \sum_{c=0}^{C} -\omega_c \boxed{\frac{2\sum_i^N p_i g_i}{\sum_i^N p_i^2 + \sum_i^N g_i^2}}$$

$N$ = Number of pixels
$p$ = Prediction
$g$ = Ground Truth

$$\omega = [\underbrace{1}_{\text{Background}}, \underbrace{2, 2, 2}_{\text{Cartilages}}, \underbrace{1, 1, 1}_{\text{Bones}}]$$

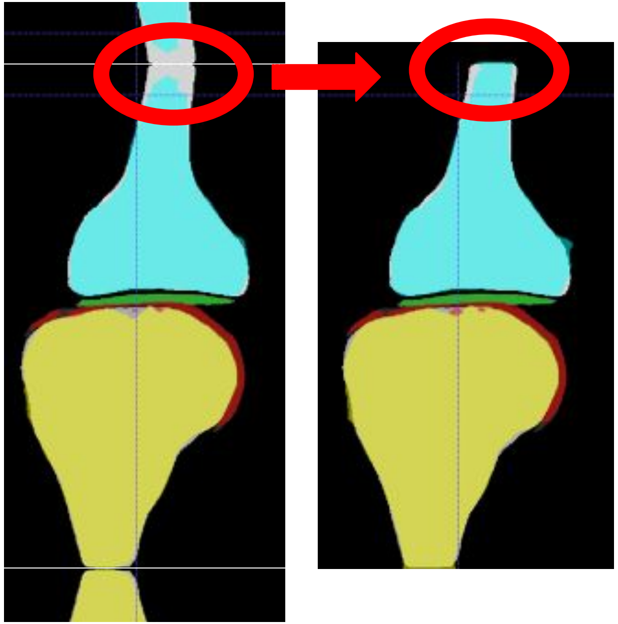
- Patch-based Training
  - Patch size: (160, 160, 48)
  - Image size: (512, 512, 196)
    - Stride: (51, 51, 16)
    - Resulting in ~27 predictions per voxel

Dice Coefficient: *F. Milletari et al, 2016*

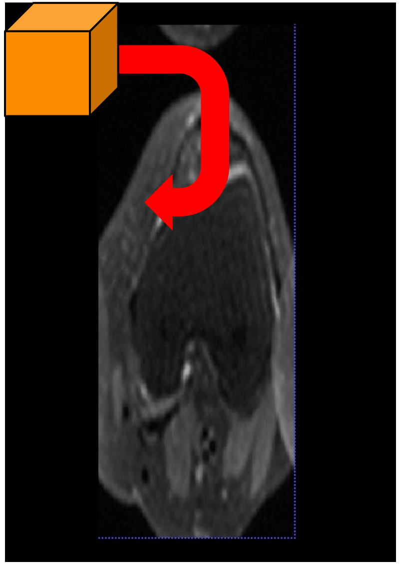
# Approach

Pre-processing - Deep Learning - **Post-processing**

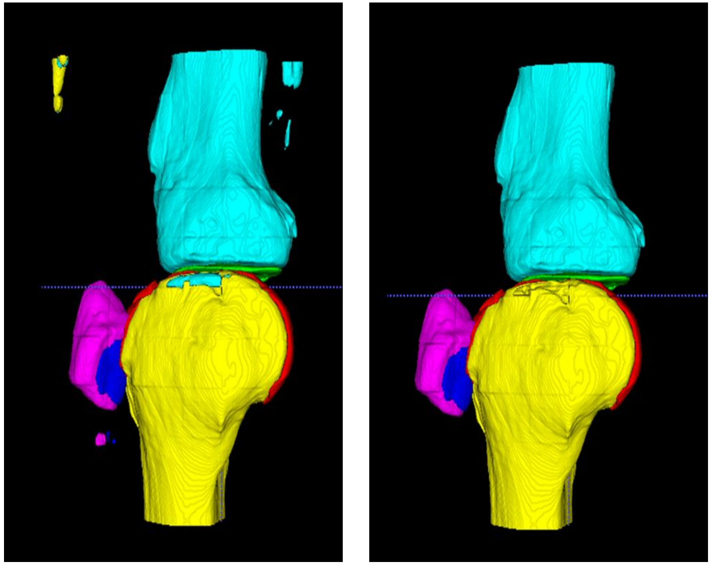
- Mirror padding



- Self-ensembling: overlapping sliding window prediction



- Connected components: Small object removal

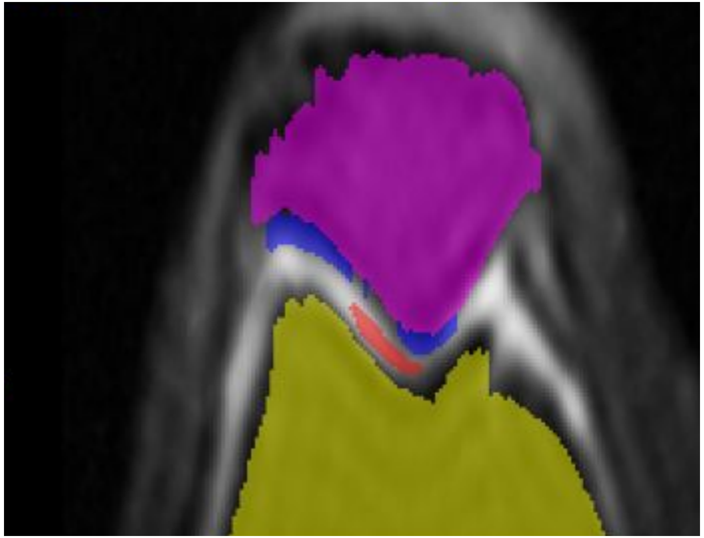


# Results

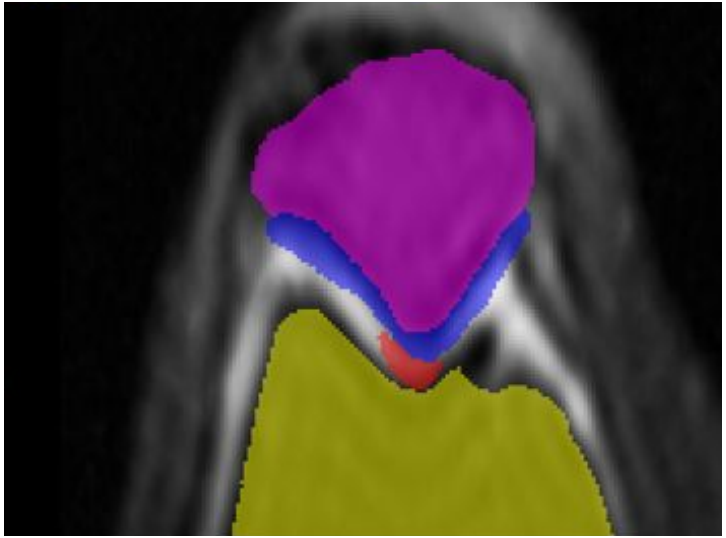
- Data subsets
  - 80% train set, 10% validation set, 10% test set
- Segmentation on **fully sampled** and 8x **undersampled** image data
  - Similar mean dice coefficients: ~0.92 DSC
- Effect of post-processing
  - Most effect on bones

# Results

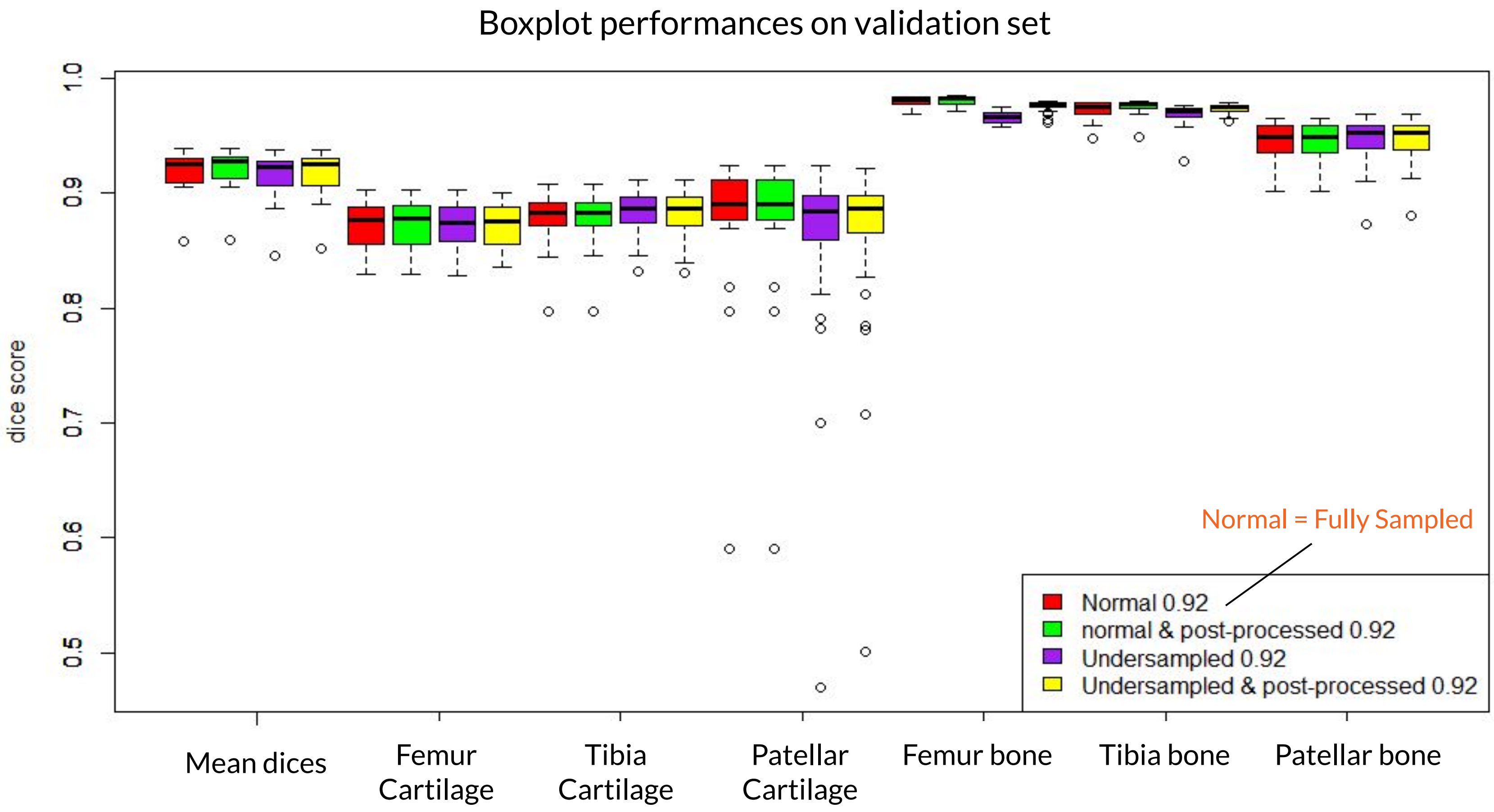
Reference Standard



Prediction



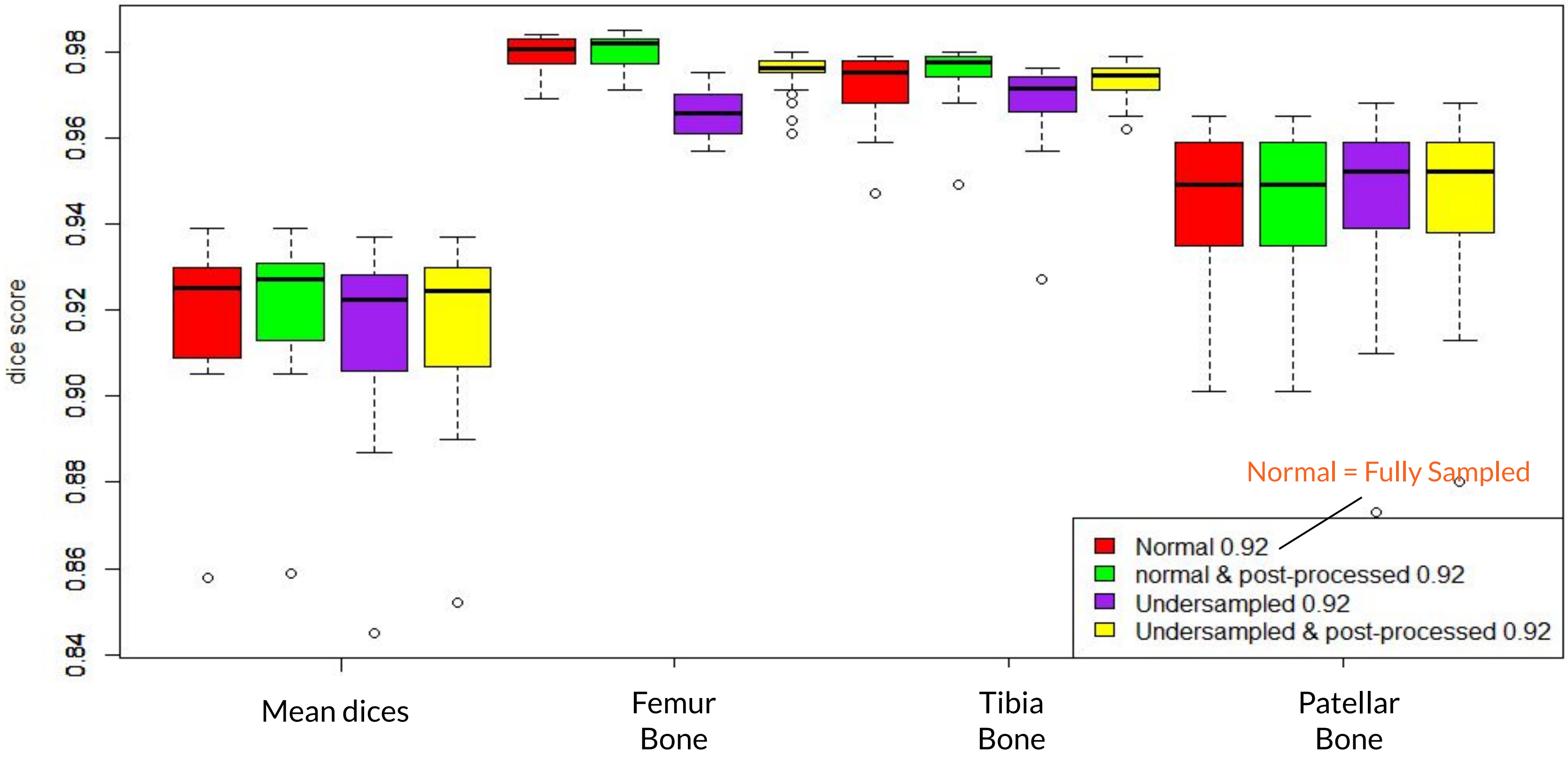
# Results



Boxplot performances on validation set
dice score
1.0
0.9
0.8
0.7
0.6
0.5
Normal = Fully Sampled
Normal 0.92
normal & post-processed 0.92
Undersampled 0.92
Undersampled & post-processed 0.92
Mean dices
Femur Cartilage
Tibia Cartilage
Patellar Cartilage
Femur bone
Tibia bone
Patellar bone

# Results

Boxplot performances on validation set



# Key Insights

- 3D Single Attention U-Net
  - Able to grasp structures in 3D
- Weighted Dice
  - More weight attributed to small scale cartilages
- Sliding Window
  - Able to process varying input sizes
- Self-ensembling
  - Increase stability of predictions
- Mirror-padding
  - Accurate predictions at image edges

# Discussion

- Fully automated end-to-end pipeline
  - Image space
  - Weighted dice
  - Patch-based self-ensembling
- Hardware limitations
  - k-space
- Clinical implementation
  - Healthy knees
  - Few labels without radiologists' complaints

# Team

Stefan Fransen
PhD Candidate

Quintin van Lohuizen
PhD Candidate

Frank Simonis
Assistant Professor

Derya Yakar
Radiologist

Thomas Kwee
Radiologist

Henkjan Huisman
Professor

**FastMRI**: "Accelerating MRI for diagnosis and interventions"

umcg Radboudumc UNIVERSITY OF TWENTE. SIEMENS Healthineers NL Health~Holland SHARED CHALLENGES, SMART SOLUTIONS

# Appendix

## Dice coefficient weights per class

- Weights = [1,1,1,1,1,1,1] # Works
- Weights = [1,2,2,2,1,1,1] # Best
- Weights = [1,10,10,10,1,1,1] # Worse
- Weights = Inversely proportional to class size # Even Worse
  - 0: 0.0005
  - 1: 0.1125
  - 2: 0.3060
  - 3: 0.4987
  - 4: 0.0064
  - 5: 0.0094
  - 6: 0.0666