Identification of Kernels in a Convolutional Neural Network

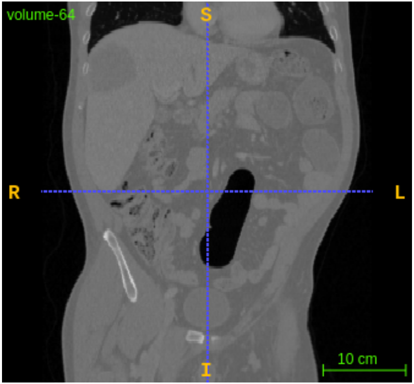
Jonas Actor

Rice University

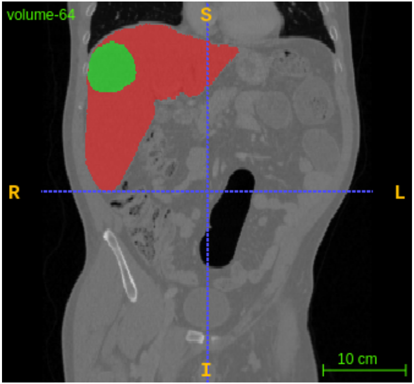
14 October 2019

jonasactor@rice.edu

Target application: medical image segmentation

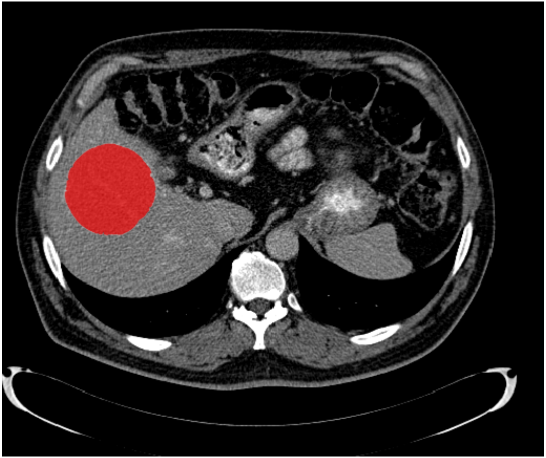


Abdominal CT scan



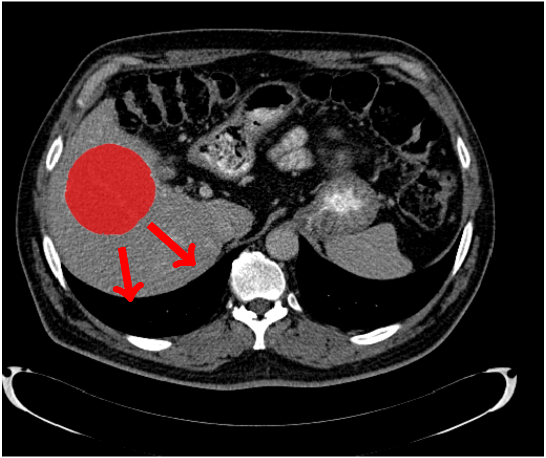
Liver and tumor segmentation

Classical Approach: Level Set Equation



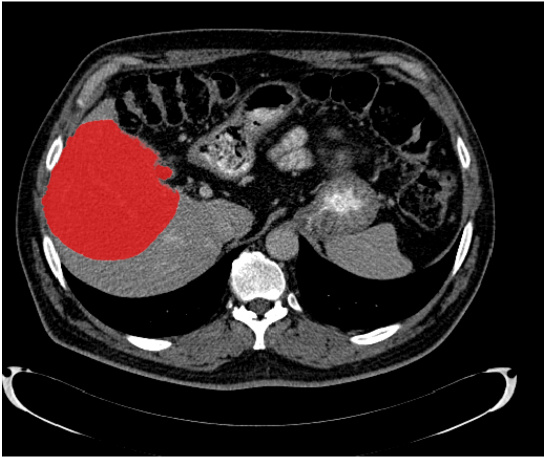
Level Set Equation: time = 200

Classical Approach: Level Set Equation



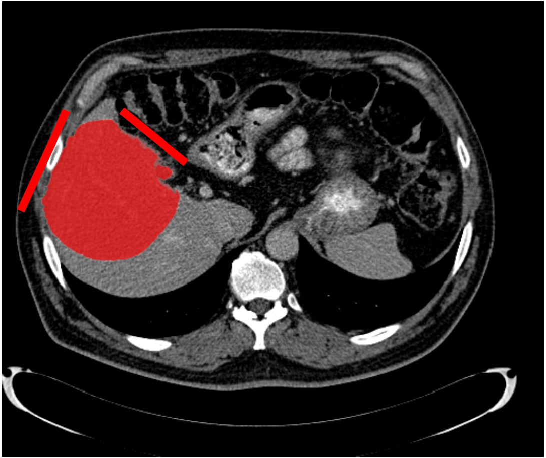
Level Set Equation: time = 200

Classical Approach: Level Set Equation



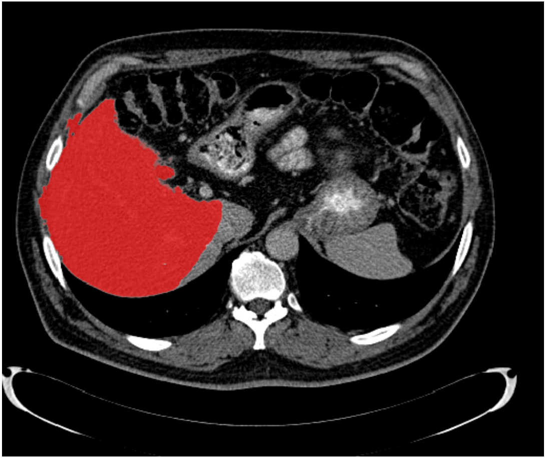
Level Set Equation: time = 500

Classical Approach: Level Set Equation



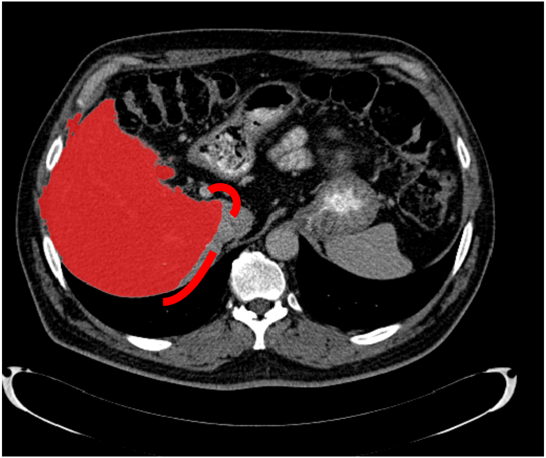
Level Set Equation: time = 500

Classical Approach: Level Set Equation



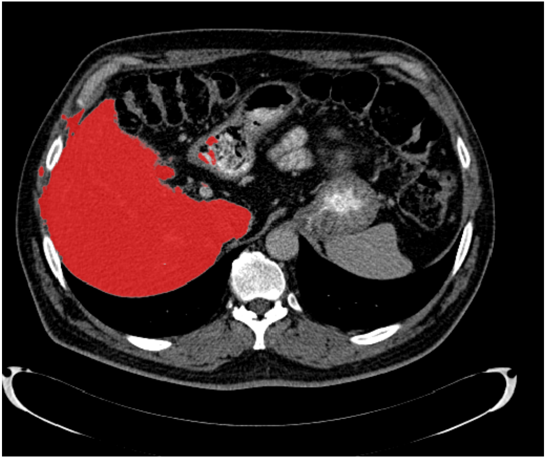
Level Set Equation: time = 1100

Classical Approach: Level Set Equation



Level Set Equation: time = 1100

Classical Approach: Level Set Equation



Level Set Equation: time = 1500

Classical Approach: Level Set Equation

- Well-established theory to analyze approximation, stability
- Upwind finite differences + fast marching method
- Semiautomated : requires initialization by user
- Works for simple problems only : relies on edge information

Motivation: LSE vs CNN

	Can analyze?	Accurate?
LSE	yes	sometimes
CNN	no	yes

Goal: accurate method we can analyze

Similarities between LSE and CNN

	LSE	CNN
Convolution	finite difference kernel	learned kernel
	$\frac{1}{h^2}\begin{bmatrix}0 & 1 & 0\\ 1 & -4 & 1\\ 0 & 1 & 0\end{bmatrix}$	$\begin{bmatrix}k_{11} & k_{12} & k_{13}\\ k_{21} & k_{22} & k_{23}\\ k_{31} & k_{32} & k_{33}\end{bmatrix}$
ReLU	upwind scheme	activation function
	$\max(0, D^+ * u) + \min(0, D^- * u)$	$\max(0, K * x + b)$

Build a CNN like a LSE solver

Level Set Equation $\longrightarrow$ Level Set Network

1. Discretize Level Set Equation
   - Explicit forward Euler in time
   - Upwind finite differences in space
2. Forward Euler $\longrightarrow$ residual skip connections
3. Upwind finite differences $\longrightarrow$ convolutions and ReLU

LSN: Results

K-Fold	LSE	LSN Test	LSN Validation	UNet
0	0.736	0.837	0.619	0.912
1	0.600	0.847	0.729	0.919
2	0.483	0.116	0.005	0.874
3	0.730	0.827	0.606	0.895
4	0.643	0.831	0.596	0.915
Avg	0.604	0.692	0.511	0.903
Avg$-\{2\}$	0.640	0.837	0.638	0.911

Table: DSC scores for each fold, from training the level set network.

Identification of Kernels

- Are CNN convolution kernels finite difference stencils?
- Are they *close* to finite difference stencils?
- What about other standard image processing kernels?

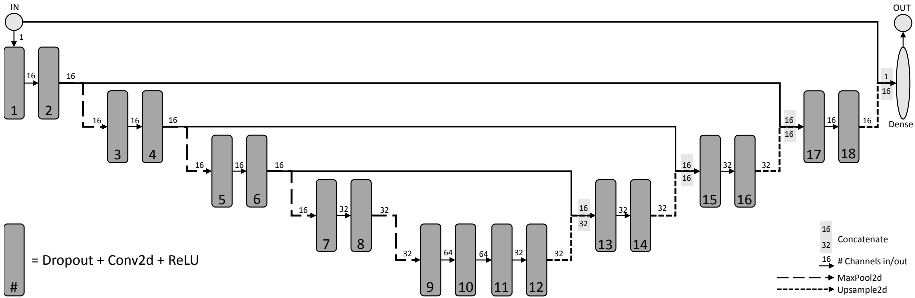
Numerical analysis kernels

- Laplacian
- Edge detection
- Identity

Image processing kernels

- Gaussian blur
- Local mean
- Sharpen

Setup of CNN



Trained on MICCAI LiTS 2017 dataset for liver segmentation

Kernel Analysis

- For each layer, separate each channel's 3×3 convolution kernel
- Flatten each 3×3 kernel into a vector $\in \mathbb{R}^9$
- Cluster with k-means
- Project down using PCA
- Project known numerical analysis and image processing kernels

Kernel Clustering Results

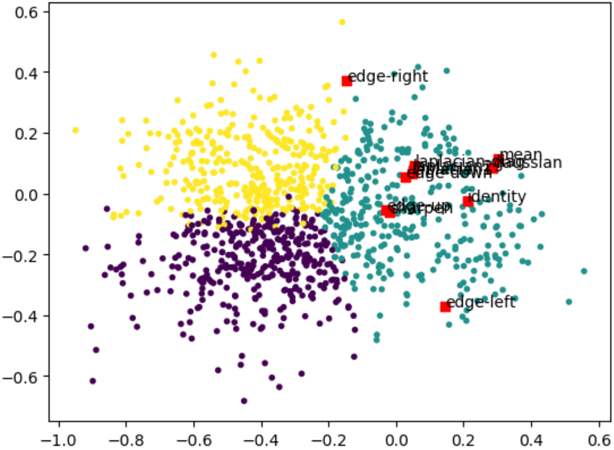


Figure: Convolution Layer 11 (encoder)

Kernel Clustering Results

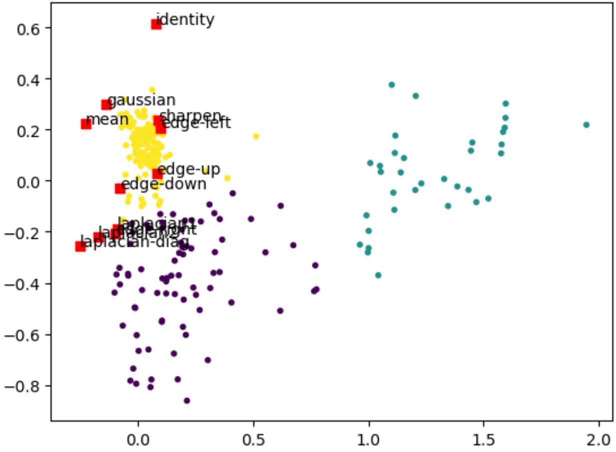


Figure: Convolution Layer 14 (decoder)

Conclusions

- Level Set Equation $\neq$ Level Set Network $\neq$ UNet
- Framework for using same operations (convolutions + ReLU) for both NNs and PDEs
- Examine how learned CNN kernels change across different layers

JAA is supported by the NLM Training Program in Biomedical Informatics & Data Science (T15LM007093), supplemented by the Ken Kennedy Institute Computer Science & Engineering Enhancement Fellowship, funded by the Rice Oil & Gas HPC Conference.