# A Method for Cross-platform Comparison of Reconstruction Kernels in CT

J. Hoffman[1] M. McNitt-Gray[1]

[1]Biomedical Physics Interdepartmental Program
University of California Los Angeles

Radiological Society of North America Annual Meeting, 2015

# Disclosures

- John Hoffman:
  - Part-time intern, Toshiba Medical Research Institute, USA, Inc.
- Michal McNitt-Gray:
  - Institutional research agreement, Siemens Healthcare
  - Past recipient, research grant support, Siemens Healthcare
  - Consultant, Toshiba America Medical Systems
  - Consultant, Samsung Electronics

# Outline

1. Introduction
   - Motivation
   - Aims

2. Kernel Extraction Approach
   - Assumptions
   - Overview

3. Evaluation
   - Methods
   - Results
   - Conclusions

# Outline

# Motivation

- Often need to standardize protocols across dissimilar scanners
- Of the parameters we control (kVp, tube output (mAs/CTDIvol), slice-thickness), *reconstruction kernel* remains problematic

# Motivation

- Often need to standardize protocols across dissimilar scanners
- Of the parameters we control (kVp, tube output (mAs/CTDIvol), slice-thickness), *reconstruction kernel* remains problematic

# Kernels

- Kernels obey some basic rules. If $G(u)$ is our kernel in the Fourier domain, and $u$ is spatial frequency, then:
  - $G(u)$ should be real and even,
  - $G(u) = |u|$ for $u$ near 0, and
  - $G(u)$ is smooth except at 0 and the Nyquist frequency.
- Kernels are otherwise somewhat "free-form"
  - Intra-manufacturer variations (name changes, scanner upgrades, etc.)
  - Inter-manufacturer variations (naming schemes, underlying kernel behavior, etc.)

# Kernels

- Kernels obey some basic rules. If $G(u)$ is our kernel in the Fourier domain, and $u$ is spatial frequency, then:
  - $G(u)$ should be real and even,
  - $G(u) = |u|$ for $u$ near 0, and
  - $G(u)$ is smooth except at 0 and the Nyquist frequency.
- Kernels are otherwise somewhat "free-form"
  - Intra-manufacturer variations (name changes, scanner upgrades, etc.)
  - Inter-manufacturer variations (naming schemes, underlying kernel behavior, etc.)

# Kernels

- Kernels obey some basic rules. If $G(u)$ is our kernel in the Fourier domain, and $u$ is spatial frequency, then:
  - $G(u)$ should be real and even,
  - $G(u) = |u|$ for $u$ near 0, and
  - $G(u)$ is smooth except at 0 and the Nyquist frequency.
- Kernels are otherwise somewhat "free-form"
  - Intra-manufacturer variations (name changes, scanner upgrades, etc.)
  - Inter-manufacturer variations (naming schemes, underlying kernel behavior, etc.)

# Kernels

- Kernels obey some basic rules. If $G(u)$ is our kernel in the Fourier domain, and $u$ is spatial frequency, then:
  - $G(u)$ should be real and even,
  - $G(u) = |u|$ for $u$ near 0, and
  - $G(u)$ is smooth except at 0 and the Nyquist frequency.
- Kernels are otherwise somewhat "free-form"
  - Intra-manufacturer variations (name changes, scanner upgrades, etc.)
  - Inter-manufacturer variations (naming schemes, underlying kernel behavior, etc.)

# Kernels

- Kernels obey some basic rules. If $G(u)$ is our kernel in the Fourier domain, and $u$ is spatial frequency, then:
  - $G(u)$ should be real and even,
  - $G(u) = |u|$ for $u$ near 0, and
  - $G(u)$ is smooth except at 0 and the Nyquist frequency.
- Kernels are otherwise somewhat “free-form”
  - Intra-manufacturer variations (name changes, scanner upgrades, etc.)
  - Inter-manufacturer variations (naming schemes, underlying kernel behavior, etc.)

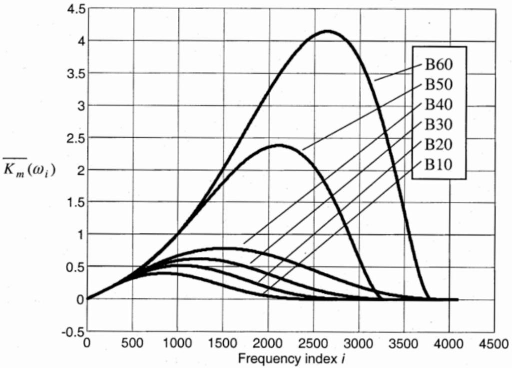



Fig. 2. Fourier transforms $\overline{K}_m(\omega_i)$ of the standard body kernels of the 4-slice CT scanner Siemens SOMATOM Volume Zoom.

Figure : Siemens CT Reconstruction kernel profiles from Volume Zoom. Source: Schaller et al. 2003

# Outline

1. Introduction
   - Motivation
   - Aims
2. Kernel Extraction Approach
   - Assumptions
   - Overview
3. Evaluation
   - Methods
   - Results
   - Conclusions

# Aim

- How can we probe beyond names like “body”, “lung”, “detail”, “B10s”, “H40f”, for some mathematical *information* about the reconstruction kernel?
- Let’s develop a method to access reconstruction kernel structure.

# Aim

- How can we probe beyond names like “body”, “lung”, “detail”, “B10s”, “H40f”, for some mathematical *information* about the reconstruction kernel?
- Let’s develop a method to access reconstruction kernel structure.

# Outline

# Assumptions

- We can perform **two sets of reconstructions from the same raw data**
  - a "test"
  - a "reference"
- Everything (algorithm, preprocessing, slice thickness, etc.) is the same ***except*** recon kernel.
- For scanner-independence, we **must know the "reference" kernel** profile in the Fourier domain.

# Assumptions

- We can perform **two sets of reconstructions from the same raw data**
  - a "test"
  - a "reference"
- Everything (algorithm, preprocessing, slice thickness, etc.) is the same ***except*** recon kernel.
- For scanner-independence, we **must know the "reference" kernel** profile in the Fourier domain.

# Outline

# The Full Kernel Extraction Process

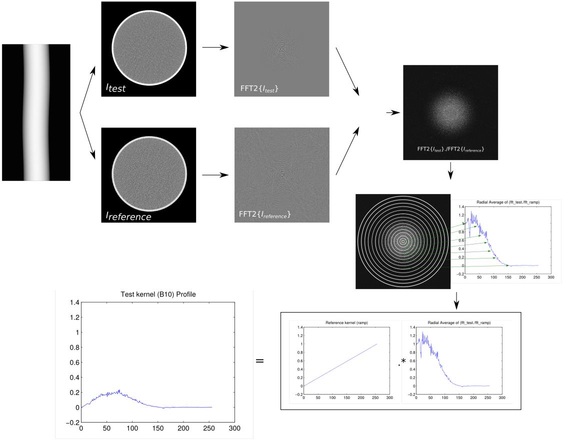



Figure : Flowchart of kernel extraction via proposed method

# Step 1: Raw data to Fourier domain image data

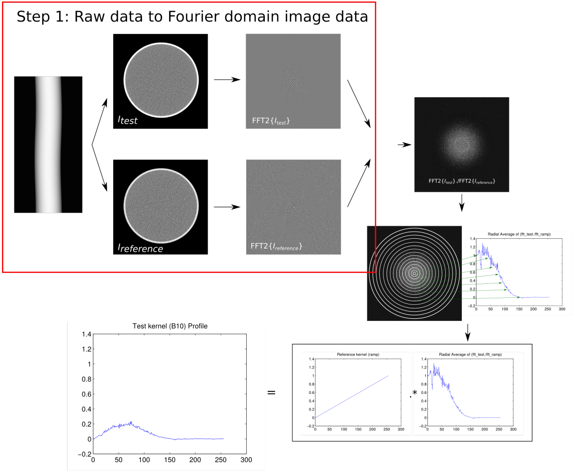
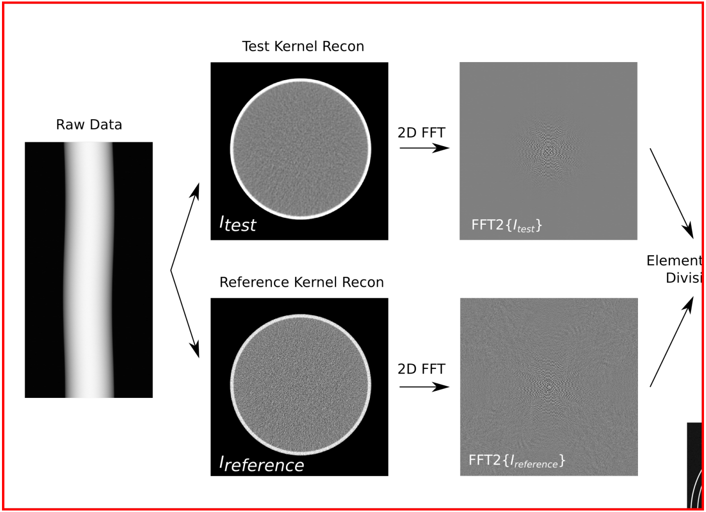
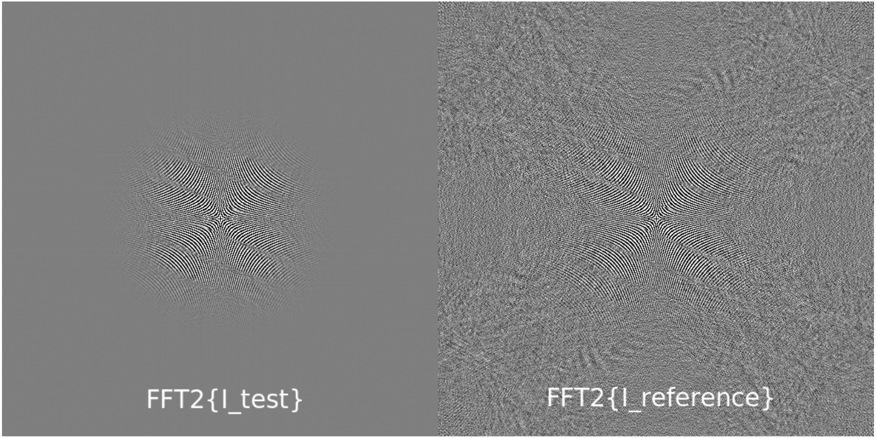



Figure : Flowchart of kernel extraction via proposed method

# Raw data to Fourier domain image data



# Fourier domain image data - detail



# Step 2: Ratio image and radial distribution

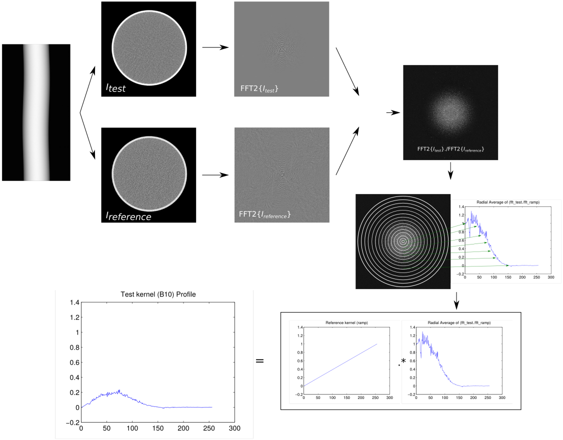



Figure : Flowchart of kernel extraction via proposed method

# Step 2: Ratio image and radial distribution

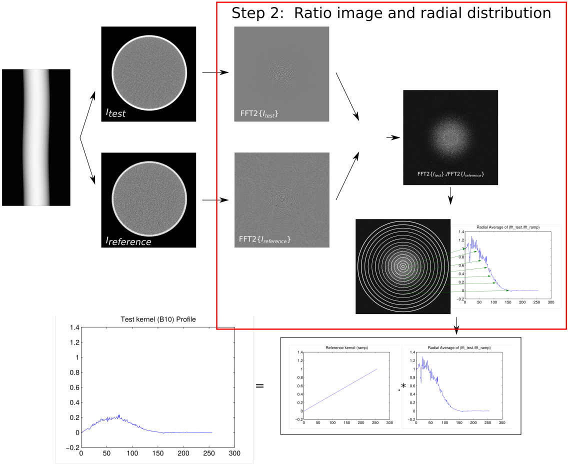
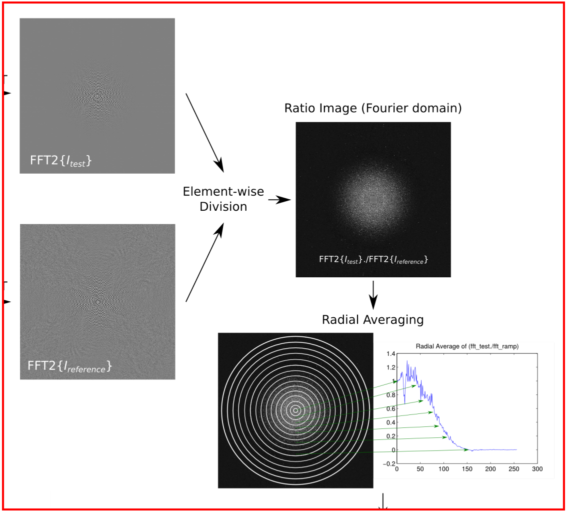



Figure : Flowchart of kernel extraction via proposed method

# Ratio image and radial distribution



# Step 3: Multiply by reference kernel (if known)

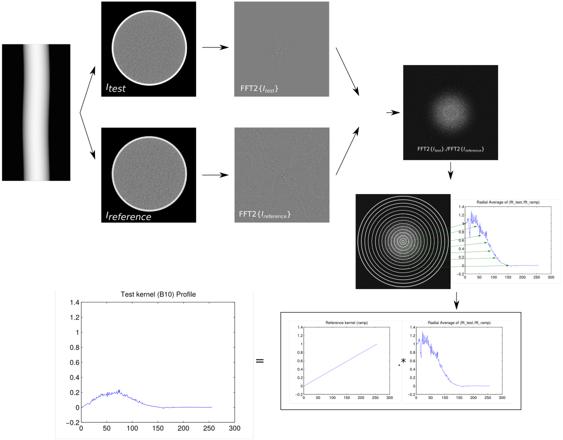



Figure : Flowchart of kernel extraction via proposed method

# Step 3: Multiply by reference kernel (if known)

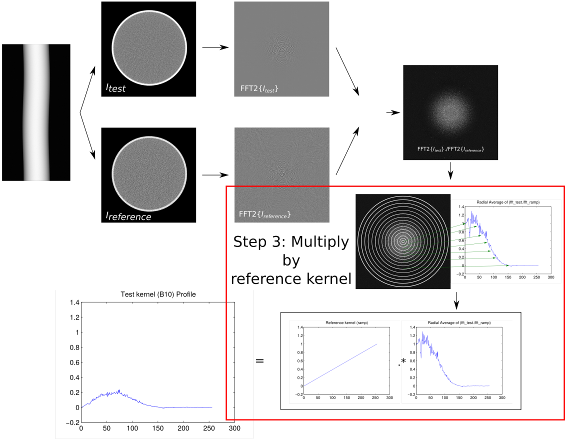
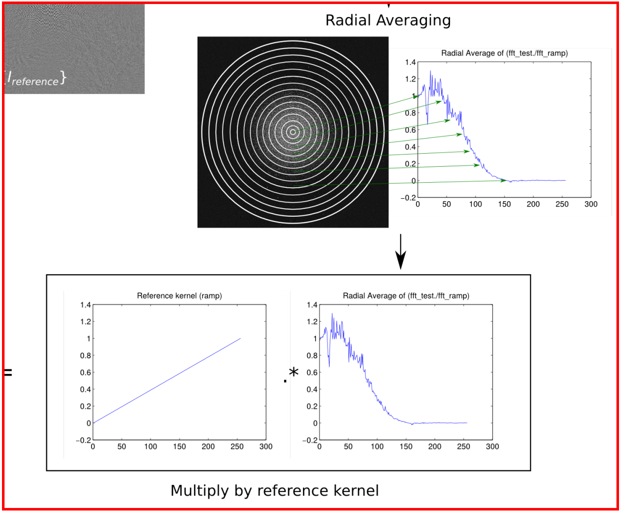



Figure : Flowchart of kernel extraction via proposed method

# Multiply by reference kernel



# And finally....

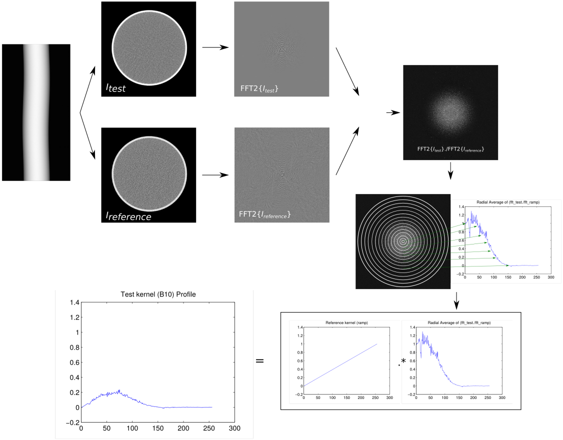



Figure : Flowchart of kernel extraction via proposed method

# Step 3 cont.: Arrive at final, absolute kernel profile

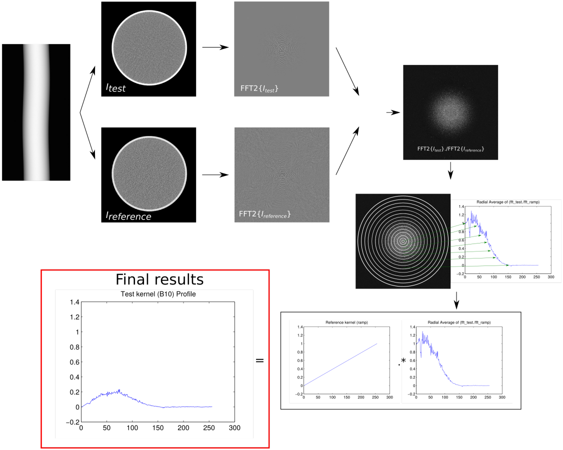
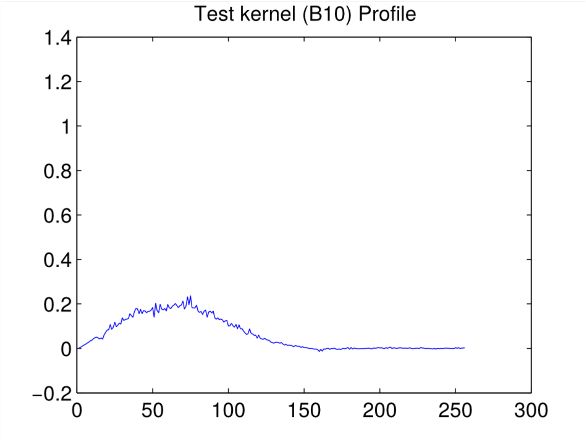



Figure : Flowchart of kernel extraction via proposed method

# Final kernel profile




=

# Outline

# Raw Data

- Method was tested on data from **Sensation 64** and **Definition AS 64** (Siemens Healthcare, Forchheim, Germany)
- For each scanner, **5 scans** through a **16 cm, centered, water phantom** were acquired with
  - 1 second rotation time
  - 64x0.6mm collimation
  - Z + Phi flying focal spots

# Reconstructions

- Each raw data file was reconstructed on the scanner with the following parameters:
  - Weighted filtered backprojection
  - Slice thickness and spacing: 0.6 mm
  - Reconstruction diameter (FOV): 250mm
  - Kernels: **B10, B20, B30, B40, B50, B60, B70, and B80** ("test" reconstructions)
- In addition, each raw data file was reconstructed using custom software, FreeCT_wFBP[1], using same parameters but with a **ramp kernel** ("reference" reconstructions).

[1] http://github.com/FreeCT/FreeCT_wFBP, submitted to MedPhys

# Reconstructions

- Each raw data file was reconstructed on the scanner with the following parameters:
  - Weighted filtered backprojection
  - Slice thickness and spacing: 0.6 mm
  - Reconstruction diameter (FOV): 250mm
  - Kernels: **B10, B20, B30, B40, B50, B60, B70, and B80** ("test" reconstructions)
- In addition, each raw data file was reconstructed using custom software, FreeCT_wFBP[1], using same parameters but with a **ramp kernel** ("reference" reconstructions).

[1] http://github.com/FreeCT/FreeCT_wFBP, submitted to MedPhys

# Analysis

- Each test image (scanner-reconstructed image) and corresponding reference (ramp image) were **analyzed using the outlined method**.
- All profiles for a given scanner and kernel were then averaged together for a final kernel profile.
- Sanity-check using scanner-specific B80 reconstruction as reference (instead of ramp kernel)

# Analysis

- Each test image (scanner-reconstructed image) and corresponding reference (ramp image) were **analyzed using the outlined method**.
- All profiles for a given scanner and kernel were then averaged together for a final kernel profile.
- Sanity-check using scanner-specific B80 reconstruction as reference (instead of ramp kernel)

# Analysis

- Each test image (scanner-reconstructed image) and corresponding reference (ramp image) were **analyzed using the outlined method**.
- All profiles for a given scanner and kernel were then averaged together for a final kernel profile.
- Sanity-check using scanner-specific B80 reconstruction as reference (instead of ramp kernel)

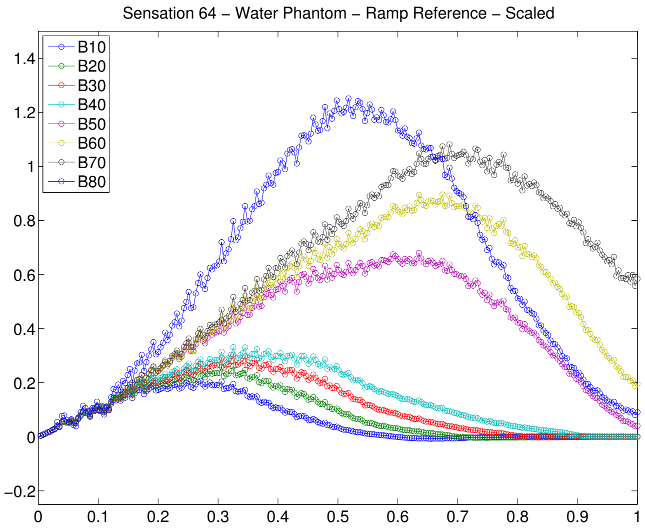
# Hypothesis: If the method works...

- We should see the **same kernel profiles between the two scanners**.

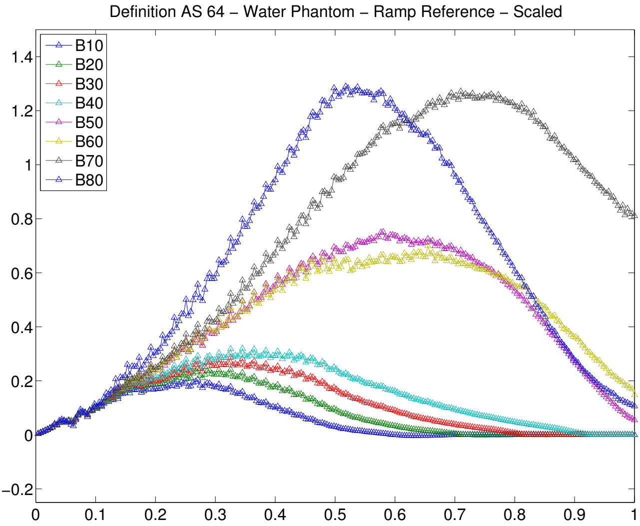
# Outline

1. Introduction
   - Motivation
   - Aims
2. Kernel Extraction Approach
   - Assumptions
   - Overview
3. Evaluation
   - Methods
   - Results
   - Conclusions

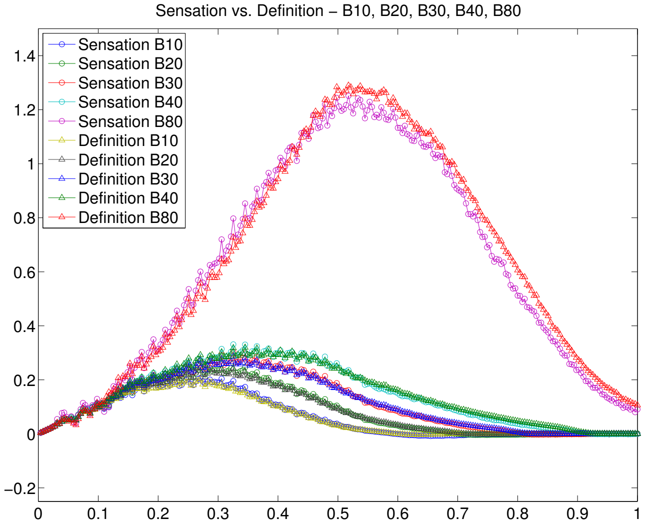
# Results: Sensation 64



# Results: Definition AS 64

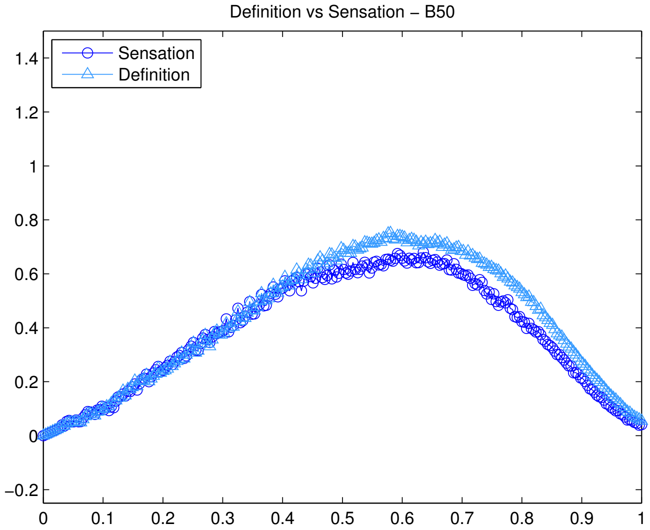


# The Good

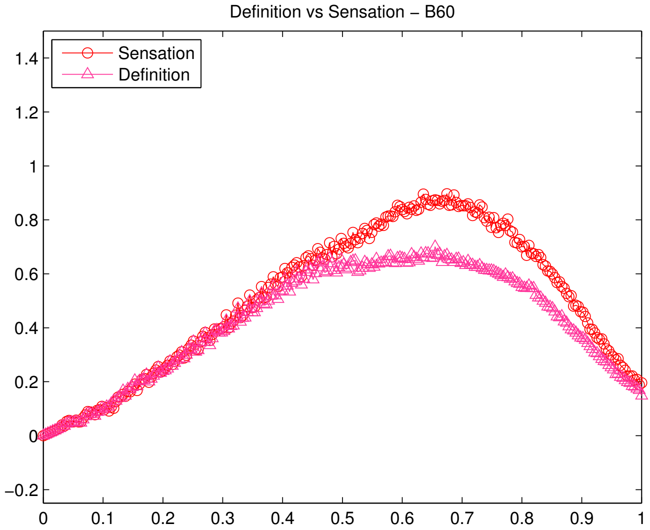


Sensation vs. Definition – B10, B20, B30, B40, B80
Sensation B10
Sensation B20
Sensation B30
Sensation B40
Sensation B80
Definition B10
Definition B20
Definition B30
Definition B40
Definition B80
1.4
1.2
1
0.8
0.6
0.4
0.2
0
−0.2
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

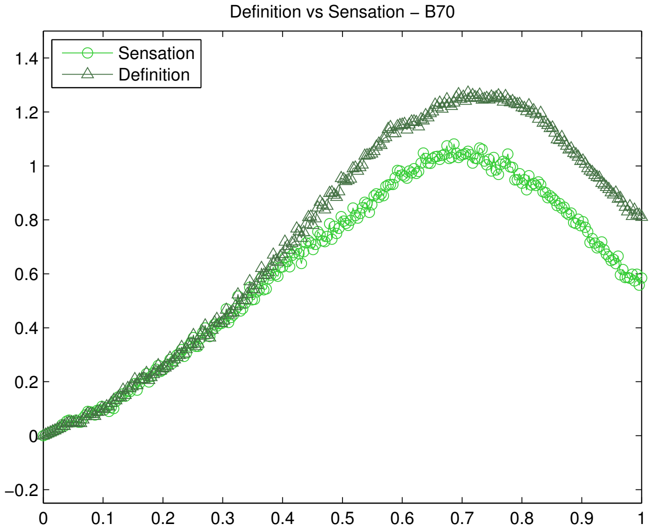
# The Not-So-Good - B50



# The Not-So-Good - B60



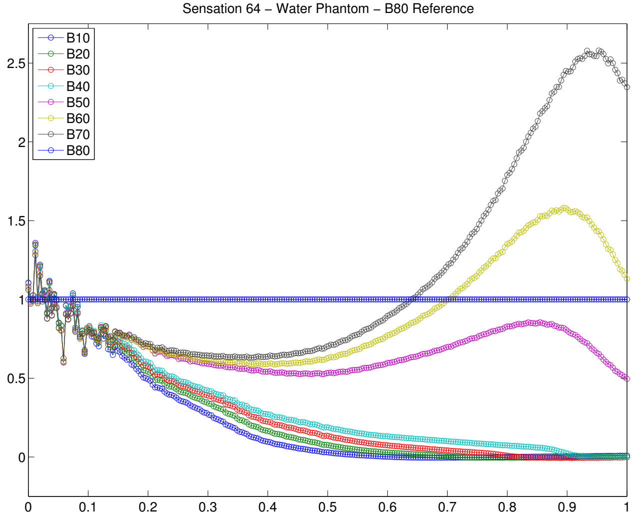
# The Not-So-Good - B70



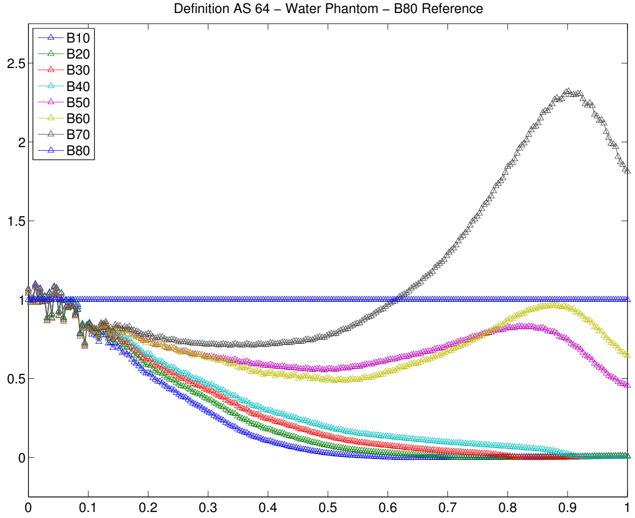
# Possible causes

- Ramp kernel reconstruction
- Reconstruction kernels are different between scanners

# Results: Sensation 64, B80 Reference



# Results: Definition AS 64, B80 Reference



# Possible causes

- Ramp kernel reconstruction
- ***Reconstruction kernels are different between scanners***

# Outline

# Conclusions

- A method to extract reconstruction kernel profiles from image data has been presented
- Possible applications include:
  - Cross-platform protocol standardization (research, clinical trials, etc.)
  - Reverse engineering

# Conclusions

- A method to extract reconstruction kernel profiles from image data has been presented
- Possible applications include:
  - Cross-platform protocol standardization (research, clinical trials, etc.)
  - Reverse engineering

# Further Work

- Different phantoms
- Scanners from other manufacturers
- Effects of FOV, slice thickness, noise, etc.
- Does matching kernels necessarily match other image performance metrics (MTF, NPS, etc.)?
- Utilizing method for quantitative imaging

# Finally...

Thank you for your interest and any questions!