Deep Image and Deep Dose Formation in CT

Marc Kachelrieß
German Cancer Research Center (DKFZ)
Heidelberg, Germany
www.dkfz.de/ct

Overview Publications in PubMed

2019 estimated for the whole year based on the values as of July 17, 2019.

Fully Connected Neural Network

- Each layer fully connects to previous layer
- Difficult to train (many parameters in \( W \) and \( b \))
- Spatial relations not necessarily preserved

\[
y(x) = f(W \cdot x + b) \quad \text{with} \quad f(x) = [f(x_1), f(x_2), \ldots] \quad \text{point-wise scalar, e.g.} \quad f(x) = x > 0 = \text{ReLU}\]

\[y^{(1)} = y^{(0)} \cdot W^{(1)} + b^{(1)}\]

\[y^{(2)} = y^{(1)} \cdot W^{(2)} + b^{(2)}\]

\[y^{(3)} = y^{(2)} \cdot W^{(3)} + b^{(3)}\]

\[y^{(4)} = y^{(3)} \cdot W^{(4)} + b^{(4)}\]

\[\ldots\]

\[y^{(n)} = y^{(n-1)} \cdot W^{(n)} + b^{(n)}\]
Convolutional Neural Network (CNN)

- Replace dense $W$ in $y(x) = f(W \cdot x + b)$ by a sparse matrix $\hat{W}$ with sparsity being of convolutional type.
- CNNs consist (mainly) of convolutional layers.
- Convolutional layers are not fully connected.
- Convolutional layers are connected by small, say 3x3, convolution kernels whose entries need to be found by training.
- CNNs preserve spatial relations to some extent.

$$D_{\text{conv}} = \sum f \left( \sum_{a,b} S_{a,b} \cdot \hat{K}^2_{a,b} \right)$$

Attention: No convolution in depth direction!

U-Net

- Input: $512 \times 512 \times 3$
- Output: $192 \times 128 \times 40$
- 3x3 Convolution, ReLU
- 2x2 Max. Pooling
- 2x2 Upsampling

$(U, \text{Ronneberger}, P. \text{Fischer}, \text{and T. \text{Brox}}. \ \text{U-Net: Convolutional networks for biomedical image segmentation}, \text{Proc. MICCAI:} 234 - 241, \text{2015.})$

Generative Adversarial Network (GAN)

- Useful, if no direct ground truth (GT) is available, the training data are unpaired, unsupervised learning.
Outline

1. Making up data
2. Noise reduction
3. Replacement of lengthy computations
4. Image reconstruction

Part 1:
Making up Data

Limited Angle Example

Image Prediction for Limited-Angle Tomography via Deep Learning with Convolutional Neural Network.
MAR Example

- Deep CNN-driven patch-based combination of the advantages of several MAR methods trained on simulated artifacts
- followed by segmentation into tissue classes
- followed by forward projection of the CNN prior and replacement of metal areas of the original sinogram
- followed by reconstruction


Sparse View Restoration Example

Very impressive, but…

Sparse CT Recon with Data Consistency Layers (DCLs)

GT
U-Net + DCL
3 Iterations
32 view FBP
2 Iterations
4 Iterations

Part 2:
Noise Reduction
Noise Removal Example 1

- 3-layer CNN uses low dose and corresponding normal dose image patches for training

Normal dose | Low dose | PS-OCS
---|---|---
KSVD | BM3D | 3-Layer CNN


Noise Removal Example 2

- Task: Reduce noise from low dose CT images.
- A conditional generative adversarial networks (GAN) is used
  - Generator $G$:
    - 3D CNN that operates on small cardiac CT sub volumes
    - Seven 3×3×3 convolutional layers yielding a receptive field of 15×15×15 voxels for each destination voxel
    - Depths (features) from 32 to 128
    - Batch norm only in the hidden layers
    - Subtracting skip connection
  - Discriminator $D$:
    - Sees either routine dose image or a generator-denosed low dose image
    - Two 3×3×3 layers followed by several 3×3 layers with varying strides
    - Feedback from $D$ prevents smoothing.
- Training:
  - Unenhanced (why?) patient data acquired with Philips Brilliance iCT 256 at 120 kV.
  - Two scans (why?) per patient, one with 0.2 mSv and one with 0.9 mSv effective dose.


Noise Removal Example 2

- $G_1$ and $G_2$ include supervised learning and thus perform only with phantom measurements.
- $G_3$ is unsupervised.
- $G_3$ is said to generate images with a more similar appearance to the routine-dose CT. Feedback from the discriminator $D$ prevents smoothing the image.
Noise Removal Example 2

Low dose image (0.2 mSv)

iDose level 3 reconstruction (0.2 mSv)

Denoised low dose image (0.2 mSv)
Noise Removal Example 2

Normal dose image (0.9 mSv)


Noise Removal Example 3

- 32 conv layers with skip connections.
- About 2 million tunable parameters in total.
- Input is a series of an axial slice of images, with a fixed number of adjacent slices in the channel/feature dimension.


Noise Removal Example 3

Low dose images (1/4 of full dose)

Noise Removal Example 3

Denoised low dose


Noise Removal Example 3

Full dose


Noise Removal Example 3

Denoised full dose

... and its Extension to DECT

<table>
<thead>
<tr>
<th>Low kV</th>
<th>High kV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full dose</td>
<td>Low dose (1/4 of full dose)</td>
</tr>
</tbody>
</table>

Noise Removal Example 4

Noise Removal Example 5

- ECG-based TCM yields cardiac phases with high noise.
- Train a cycle GAN that learns from the low noise phases to remove noise in the high noise phases.
- 50 patient cases used for training.
- Nice results!

Based on a deep CNN
- Trained to restore low-dose CT data to match the properties of FIRST, the model-based IR of Canon.
- FIRST is applied to high-dose CT images to obtain a high-fidelity training target.

Noise Removal Example 6
Canon’s AiCE

Noise Removal Example 7
GE's True Fidelity
- Based on a deep CNN
- Trained to restore low-dose CT data to match the properties of Veo, the model-based IR of GE.
- No information can be obtained in how the training is conducted for the product implementation.
No Low Noise Images Required to Train Denoising Networks!

• Estimation can be regarded as ML estimation by interpreting the loss function as the negative log likelihood.
• On expectation, the estimate remains unchanged if we replace the targets with random numbers whose expectations match the targets.
• Input-conditioned target distributions \( p(y|x) \) can be replaced with arbitrary distributions that have the same conditional expected values.
• Consequently, we may corrupt the training targets of a neural network with zero-mean noise without changing what the network learns.
• Useful for image restoration tasks where the expectation of the corrupted input data is the clean target that we seek to restore.
• Denoising possible if at least two realizations of each image are available.

Noise Removal Example 8 (Training on Noisy CT Targets)

Part 3:
Replacement of Lengthy Computations
Fast Physics

Empirical Shading Correction: ScatterNet

- Net to convert CBCT log (why?) rawdata into artifact-free data.
- Net architecture:
  - Small receptive field spectrum converter block adapts the attenuation values.
    - Residual U-Net then follows to account for scatter.
- Pixel-wise loss function comparing the corrected CBCT projections with those of the reference shading correction method.
- Reference shading correction method:
  - Use data from a clinical CT scan as an artifact-free prior.
  - Intensity domain frequency split between planning CT and CBCT.
    - Deformably register planning CT onto CBCT and forward project and exponentiate to obtain "ideal" intensity data
    - Scale CBCT intensities to match the prior CT intensities
  - Corrected intensities = LP(forward proj. CT)+HP(scaled uncorr. CBCT)
- ScatterNet replaces the previous correction method and thus speeds up computation and does not make use of the planning CT.

Deep Scatter Estimation

Scatter Correction

Scatter suppression
- Anti-scatter grids
- Collimators
- ...

Scatter estimation
- Monte Carlo simulation
- Kernel-based approaches
- Boltzmann transport
- Primary modulation
- Beam blockers
- ...

- Simulation of photon trajectories according to physical interaction probabilities.
- Simulating a large number of photon trajectories well approximates the actual scatter distribution.

1 to 10 hours per tomographic data set
Deep Scatter Estimation (DSE)

Train a deep convolutional neural network (CNN) to estimate scatter using a function of the acquired projection data as input.

Deep Scatter Estimation
Network architecture & scatter estimation framework

Training the DSE Network

- Simulation of 6000 projections using different heads and acquisition parameters (80 kV, ..., 140 kV in steps of 20 kV).
- Splitting into 80% training and 20% validation data.
- Mean S/P = 0.9
- 90th percentile S/P = 1.32
- Training minimizes MSE pixel-wise loss on a GeForce GTX 1080 for 80 epochs.
Testing of the DSE Network for Simulated Data (at 120 kV)

- Application of the DSE network to predict scatter for simulated data of a head (different from training data).

Ref 1: Kernel-Based Scatter Estimation

- Kernel-based scatter estimation:
  - Estimation of scatter by a convolution of the scatter source term with a scatter propagation kernel:
    \[ \hat{s}(u,v) = \int \sum_{i} e^{-i(k_{i} \cdot \cos \theta_{i})} \]
Hybrid scatter estimation\textsuperscript{2}:

Estimation of scatter by a convolution of the scatter source term \( f_s \) with a scatter propagation kernel \( f_s(p) \) for each direction \( p \):

\[
\hat{i}_s = \sum_i \left( f_s(p) \ast \overline{f_s}(p) \right) = \sum_i \left( \int f_s(p) \overline{f_s}(p') dp' \right)
\]

\[f_s(p) = \sum_c \left( i_c \ast f_c \right)
\]

Open parameters: \( C_1, C_2, C_3, C_4 \)

Samples of the test data set

Results on Simulated Projection Data

DSE trained to estimate scatter from primary plus scatter: High accuracy

DSE, in its present form, needs to see scatter in its input data!
**Results on Simulated Projection Data**

<table>
<thead>
<tr>
<th>View</th>
<th>Scatter ground truth (GT)</th>
<th>Hybrid-GT</th>
<th>DSE-GT</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1</td>
<td>14.1%</td>
<td>7.2%</td>
<td>1.2%</td>
</tr>
<tr>
<td>#2</td>
<td>14.1%</td>
<td>7.2%</td>
<td>1.2%</td>
</tr>
<tr>
<td>#3</td>
<td>14.1%</td>
<td>7.2%</td>
<td>1.2%</td>
</tr>
<tr>
<td>#4</td>
<td>14.1%</td>
<td>7.2%</td>
<td>1.2%</td>
</tr>
<tr>
<td>#5</td>
<td>14.1%</td>
<td>7.2%</td>
<td>1.2%</td>
</tr>
</tbody>
</table>

DSE trained to estimate scatter from primary plus scatter: High accuracy

---

**Reconstructions of Simulated Data**

- Ground Truth
- No Correction
- Kernel-Based Scatter Estimation
- Hybrid Scatter Estimation
- Deep Scatter Estimation

---

**Testing of the DSE Network for Measured Data (120 kV)**

- Measurement of a head phantom at our in-house table-top CT.
- Slit scan measurement serves as ground truth.

---

DSE, in its present form, needs to see scatter in its input data!
Reconstructions of Measured Data

No Correction  Kernel-Based Scatter Estimation  Hybrid Scatter Estimation  Deep Scatter Estimation

C = 0 HU, W = 1000 HU


Generalization to Different Anatomical Regions

<table>
<thead>
<tr>
<th></th>
<th>Head</th>
<th>Thorax</th>
<th>Abdomen</th>
</tr>
</thead>
<tbody>
<tr>
<td>Head</td>
<td>1.2</td>
<td>21.1</td>
<td>32.7</td>
</tr>
<tr>
<td>Thorax</td>
<td>8.8</td>
<td>1.5</td>
<td>9.1</td>
</tr>
<tr>
<td>Abdomen</td>
<td>11.9</td>
<td>10.9</td>
<td>1.3</td>
</tr>
<tr>
<td>All data</td>
<td>1.8</td>
<td>1.4</td>
<td>1.4</td>
</tr>
</tbody>
</table>

Values shown are the mean absolute percentage errors (MAPEs) of the testing data.

To learn why MC fails at truncated data and what significant efforts are necessary to cope with that situation see [Kachelrieß et al. Effect of detruncation on the accuracy of MC-based scatter estimation in truncated CBCT. Med. Phys. 45(8):3574-3590, August 2018].

A simple detruncation was applied to the raw data before reconstruction. Images were clipped to the FOM before display.

C = -200 HU, W = 1000 HU.

Deep scatter estimation (DSE) for truncated cone-beam CT (CBCT). RSNA 2018.

### Conclusions on DSE

- DSE needs about 20 ms per projection. It is a fast and accurate alternative to Monte Carlo (MC) simulations.
- DSE outperforms kernel-based approaches in terms of accuracy and speed.
- **Interesting observations**
  - DSE can estimate scatter from a single (I) x-ray image.
  - DSE can accurately estimate scatter from a primary-scatter image.
  - DSE cannot accurately estimate scatter from a primary only image.
  - DSE may thus outperform MC even though DSE is trained with MC.
- DSE is not restricted to reproducing MC scatter estimates.
- DSE can rather be trained with any other scatter estimate, including those based on measurements.

### Estimation of Dose Distributions

- Useful to study dose reduction techniques
  - Tube current modulation
  - Prefiltration and shaped filtration
  - Tube voltage settings
- **Useful to estimate patient dose**
  - Risk assessment requires segmentation of the organs (difficult)
  - Often semianthropomorphic patient models take over
  - The infamous k-factors that convert DLP into D_eff are derived this way, e.g. k_chest = 0.014 mSv/mGy/cm
- Useful for patient-specific CT scan protocol optimization
- **However:** Dose estimation does not work in real time!

### Influence of Bowtie Filter

- Commercial CT-scanners are usually equipped with a bowtie filter in order to optimize the patient dose distribution.
- Monte-Carlo dose calculations or statistical reconstruction algorithms require exact knowledge of the bowtie filter.
- The shape as well as the composition of the bowtie filter is usually not disclosed by the CT vendors.
Patient-Specific Dose Estimation

• Accurate solutions:
  – Monte Carlo (MC) simulation\(^1\), gold standard, stochastic LBTE solver
  – Analytic linear Boltzmann transport equation (LBTE) solver\(^2\)
  \(\rightarrow\) Accurate but computationally expensive

• Fast alternatives:
  – Application of patient-specific conversion factors to the DLP\(^3\).
  – Application of look-up tables using MC simulations of phantoms\(^4\).
  – Analytic approximation of CT dose deposition\(^5\).
  \(\rightarrow\) Fast but less accurate


Deep Dose Estimation (DDE)

• Combine fast and accurate CT dose estimation using a deep convolutional neural network.
• Train the network to reproduce MC dose estimates given the CT image and a first-order dose estimate.

Training and Validation

• Simulation of 1440 circular dual-source CT scans (±0.6 mm, FOM\(_A\) = 50 cm, FOM\(_B\) = 32 cm) of thorax, abdomen, and pelvis using 12 different patients.
• Simulation with and without bowtie.
• No data augmentation
• Reconstruction on a 512x512x96 grid with 1 mm voxel size, followed by 2x2x2 binning for dose estimation.
• 9 patients were used for training and 3 for testing.
• DDE was trained for 300 epochs on an Nvidia Quadro P6000 GPU using a mean absolute error pixel-wise loss, the Adam optimizer, and a batch size of 4.
• The same weights and biases were used for all cases.
Results
Thorax, tube A, 120 kV, with bowtie

CT image  First order dose

<table>
<thead>
<tr>
<th>Method</th>
<th>DDE</th>
<th>MC</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>1 h</td>
<td>0.25 s</td>
</tr>
<tr>
<td>GPU</td>
<td>20 h</td>
<td>5 s</td>
</tr>
</tbody>
</table>

MC uses 16 CPU kernels
DDE uses one Nvidia Quadro P600 GPU
DDE training took 15 h for 300 epochs, 1440 samples, 48 slices per sample

Relative error:
\[ C = 0\% \]
\[ W = 40\% \]


Results
Thorax, tube A, 120 kV, no bowtie

CT image  First order dose

<table>
<thead>
<tr>
<th>Method</th>
<th>DDE</th>
<th>MC</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>1 h</td>
<td>0.25 s</td>
</tr>
<tr>
<td>GPU</td>
<td>20 h</td>
<td>5 s</td>
</tr>
</tbody>
</table>

MC uses 16 CPU kernels
DDE uses one Nvidia Quadro P600 GPU
DDE training took 15 h for 300 epochs, 1440 samples, 48 slices per sample

Relative error:
\[ C = 0\% \]
\[ W = 40\% \]


Results
Thorax, tube B, 120 kV, no bowtie

CT image  First order dose

<table>
<thead>
<tr>
<th>Method</th>
<th>DDE</th>
<th>MC</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>1 h</td>
<td>0.25 s</td>
</tr>
<tr>
<td>GPU</td>
<td>20 h</td>
<td>5 s</td>
</tr>
</tbody>
</table>

MC uses 16 CPU kernels
DDE uses one Nvidia Quadro P600 GPU
DDE training took 15 h for 300 epochs, 1440 samples, 48 slices per sample

Relative error:
\[ C = 0\% \]
\[ W = 40\% \]

Results

Abdomen, tube A, 120 kV, with bowtie

CT image  First order dose

<table>
<thead>
<tr>
<th>Method</th>
<th>Dose</th>
<th>Time</th>
<th>Relative error</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC</td>
<td>1 h</td>
<td>0.25 s</td>
<td>20 h 5 s</td>
</tr>
<tr>
<td>DDE</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

MC uses 16 CPU kernels
DDE uses one Nvidia Quadro P600 GPU
DDE training took 74 h for 300 epochs, 1440 samples, 48 slices per sample

Relative error

C = 0%
W = 40%


Results

Abdomen, tube A, 120 kV, no bowtie

CT image  First order dose

<table>
<thead>
<tr>
<th>Method</th>
<th>Dose</th>
<th>Time</th>
<th>Relative error</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC</td>
<td>1 h</td>
<td>0.25 s</td>
<td>20 h 5 s</td>
</tr>
<tr>
<td>DDE</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

MC uses 16 CPU kernels
DDE uses one Nvidia Quadro P600 GPU
DDE training took 74 h for 300 epochs, 1440 samples, 48 slices per sample

Relative error

C = 0%
W = 40%


Results

Abdomen, tube B, 120 kV, no bowtie

CT image  First order dose

<table>
<thead>
<tr>
<th>Method</th>
<th>Dose</th>
<th>Time</th>
<th>Relative error</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC</td>
<td>1 h</td>
<td>0.25 s</td>
<td>20 h 5 s</td>
</tr>
<tr>
<td>DDE</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

MC uses 16 CPU kernels
DDE uses one Nvidia Quadro P600 GPU
DDE training took 74 h for 300 epochs, 1440 samples, 48 slices per sample

Relative error

C = 0%
W = 40%

Results
Pelvis, tube A, 120 kV, with bowtie

CT Image  First order dose

<table>
<thead>
<tr>
<th></th>
<th>MC</th>
<th>DDE</th>
</tr>
</thead>
<tbody>
<tr>
<td>48 slices</td>
<td>1 h</td>
<td>0.25 s</td>
</tr>
<tr>
<td>total</td>
<td>20 h</td>
<td>5 s</td>
</tr>
</tbody>
</table>

MC uses 16 CPU threads
DDE uses one Nvidia Quadro P600 GPU
DDE training took 74 h for 300 epochs, 1440 samples, 48 slices per sample

Relative error

\[ C = 0\% \]
\[ W = 40\% \]

MC ground truth
DDE

Pelvis, tube A, 120 kV, no bowtie

CT Image  First order dose

<table>
<thead>
<tr>
<th></th>
<th>MC</th>
<th>DDE</th>
</tr>
</thead>
<tbody>
<tr>
<td>48 slices</td>
<td>1 h</td>
<td>0.25 s</td>
</tr>
<tr>
<td>total</td>
<td>20 h</td>
<td>5 s</td>
</tr>
</tbody>
</table>

MC uses 16 CPU threads
DDE uses one Nvidia Quadro P600 GPU
DDE training took 74 h for 300 epochs, 1440 samples, 48 slices per sample

Relative error

\[ C = 0\% \]
\[ W = 40\% \]

MC ground truth
DDE

Pelvis, tube B, 120 kV, no bowtie

CT Image  First order dose

<table>
<thead>
<tr>
<th></th>
<th>MC</th>
<th>DDE</th>
</tr>
</thead>
<tbody>
<tr>
<td>48 slices</td>
<td>1 h</td>
<td>0.25 s</td>
</tr>
<tr>
<td>total</td>
<td>20 h</td>
<td>5 s</td>
</tr>
</tbody>
</table>

MC uses 16 CPU threads
DDE uses one Nvidia Quadro P600 GPU
DDE training took 74 h for 300 epochs, 1440 samples, 48 slices per sample

Relative error

\[ C = 0\% \]
\[ W = 40\% \]

MC ground truth
DDE

Conclusions on DDE

- As shown, DDE works well with 360° circle scans.
- What is not shown in this presentation is that DDE can be trained to provide accurate dose predictions:
  - for sequence scans
  - for partial scans (less than 360°)
  - for spiral scans
  - for different tube voltages
  - for scans with and without bowtie filtration
  - for scans with tube current modulation
- In practice it may therefore be not necessary to perform separate training runs for these cases.
- Thus, accurate real-time patient dose estimation may become feasible with DDE.


Best Paper within Machine Learning at ECR 2019!

Part 4: Image Reconstruction

Often “Just” Image Restoration

- Speeding up iterative reconstruction by training a CNN to convert an FBP image into an iterative image
  - Canon's AiCE algorithm
  - GE's True Fidelity algorithm
  - plus a few more algorithms proposed in the literature
- Noise reduction by training, e.g. a mapping from low dose to high dose images
  - many examples in the literature, some in this presentation
- Artifact reduction in image domain
  - many examples in the literature, one shown in this presentation
- ...
Sometimes “Real” Image Reconstruction

- Networks employing data consistency layers
- Networks including backprojection layers
- Learning of backprojectors
- End-to-end training from sinogram to image
- Unrolled iterative reconstruction with learned priors
- …

Sparse CT Recon with Data Consistency Layers (DCLs)


Variational Network-Based Image Reconstruction

Conclusions on Deep CT

- Machine learning will play a significant role in CT image formation.
- High potential for:
  - Artifact correction
  - Noise and dose reduction
  - Real-time dose assessment (also for RT)
- Care has to be taken:
  - Underdetermined acquisition, e.g., sparse view or limited angle CT, require the net to make up information!
  - Nice looking images do not necessarily represent the ground truth.
  - Data consistency layers and variational networks with raw data access may ensure that the information that is made up is consistent with the measured data.

Thank You!

The 6th International Conference on Image Formation in X-Ray Computed Tomography
August 3 - August 7, 2020, Regensburg, Germany • www.ct-meeting.org