Deep-learning uncertainty estimation for data-consistent breast tomosynthesis reconstruction

Arnaud Quillent, Vincent Bismuth, Isabelle Bloch, Saïd Ladjal, Christophe Kervazo

DEEP-LEARNING UNCERTAINTY ESTIMATION FOR DATA-CONSISTENT BREAST TOMOSYNTHESIS RECONSTRUCTION

Arnaud Quillent [1,2] *Vincent Bismuth* [2] *Isabelle Bloch* [1,3] *Christophe Kervazo* [1] *Saïd Ladjal* [1]

[1] LTCI, Télécom Paris, Institut Polytechnique de Paris, Palaiseau, France
[2] GE HealthCare, Buc, France
[3] Sorbonne Université, CNRS, LIP6, Paris, France

ABSTRACT

Digital Breast Tomosynthesis (DBT) is an X-ray modality enabling to reconstruct 3D volumes in the context of breast cancer screening. However, because of the limited angle and sparse view constraints, artefacts emerge in the reconstructions and greatly reduce their quality. In a previous work, we proposed a post-processing deep learning reconstruction pipeline for DBT that is trained using synthetic data. Owing to the geometrical limitations of the acquisition device, the amount of information to extrapolate is important and the neural network could inevitably commit errors. As such, the reconstructed volumes are not completely reliable, and exact consistency with the measurements is not guaranteed. In this study, we first propose two methods to estimate the uncertainty of the model reconstructions, and show that the result can be used as a proxy of the true error. Secondly, we explore the minimisation of a data consistency term constrained by the predicted uncertainty, in order to mitigate the network errors. We demonstrate experimentally that this approach enhances the quality of reconstruction as compared to reintroducing projections information without constraint.

***Index Terms*—** Deep learning, tomosynthesis, reconstruction, uncertainty, inverse problem.

1. INTRODUCTION

Digital Breast Tomosynthesis (DBT) DBT [1] is an X-ray imaging technique that is primarily employed in the context of breast cancer screening. During a DBT examination, several low-dose cone-beam acquisitions are performed from different angles, which are subsequently used to reconstruct a 3D volume. However, a number of geometrical constraints impede the quality of the reconstructed images [2, 3]. Indeed, the rotation of the X-ray source is restricted to a narrow angular range (limited angle), and the number of acquired projections does not exceed a dozen (sparse view). Consequently, the resolution along the vertical detector-to-source axis (z-axis) is severely reduced: objects are spread through multiple horizontal planes and tend to blend as they vanish at a slow pace. Furthermore, many artefacts appear in the images, forming streaks or replications depending on the anatomical plane considered. In this work, we simulate the use of a DBT system thanks to GPU-based algorithms, and acquire 9 projections on a 25° angular range.

Mathematical framework Let $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{y} \in \mathbb{R}^m$ be the vectorised versions of the 3D object to image and the corresponding DBT projections. The X-ray acquisition can be modelled as $\mathbf{y} = \mathbf{A}\mathbf{x} + \boldsymbol{\eta}$, with $\mathbf{A} \in \mathbb{R}^{m \times n}$ the projection operator and $\boldsymbol{\eta} \in \mathbb{R}^m$ a noise component [2, 3]. DBT reconstruction is an inverse problem: from the measurements $\mathbf{y}$, the goal is to retrieve $\mathbf{x}$. However, because of the physical limitations of the system, $\mathbf{y}$ does not carry enough information to retrieve a perfect estimation of $\mathbf{x}$. Thereby, this problem is severely ill-posed, leading to an infinite number of possible solutions that are of unequal quality.

Contributions In [4], we proposed a post-processing approach in which a deep neural network (NN) is used to remove the artefacts present in an initial filtered back-projection reconstruction. Yet, because of the geometrical setting of the DBT system, the neural network has to recreate a lot of information not present in the data. Therefore, the prediction given by the neural network, despite its potential accuracy, cannot be fully trusted. Furthermore, maintaining consistency with the measurements becomes challenging when a large portion of the data is extrapolated ($n \gg m$) as the model errors in the reconstructed volume would modify the information present in the measurements once re-projected. Yet, this consistency is crucial as the acquired projections are the only indisputable data. For this reason, we compare in this paper two methods enabling the neural network to evaluate its own reliability, taking into account both epistemic and aleatoric uncertainties. Such estimates are then used in an easily understandable iterative post-processing step in order to improve the data consistency of the reconstructed volume and correct their potential errors, while limiting the reintroduction of artefacts. Eventually, our method is experimentally validated on simulated datasets. To our best knowledge, deep learning reconstruction with uncertainty quantification for an X-ray imaging geometry as constrained as the DBT system used here has never been addressed before.

Related works Uncertainty [5] is generally broken down into *epistemic* and *aleatoric* terms. The former stems from the model optimisation, while the latter is part of the data themselves and thus cannot be reduced. To model *aleatoric* uncertainty, a common choice in the literature is to consider that the error of pixel j from image $\mathbf{i}$, noted $[\boldsymbol{\epsilon}_\mathbf{i}]_j$, follows a certain distribution with zero-mean and a pixel-dependent scale parameter $[\hat{\mathbf{s}}_\mathbf{i}]_j$ (a property called heteroscedasticity) [6]: $[\boldsymbol{\epsilon}_\mathbf{i}]_j \sim \mathcal{P}_\epsilon(0, [\hat{\mathbf{s}}_\mathbf{i}]_j)$. Regarding *epistemic* uncertainty, the majority of methods rely on getting several plausible output images from a single input: their mean is taken as the prediction while their standard deviation is the uncertainty. To this aim, the authors of [7] proposed to use *deep ensembles*: a certain number of deep networks are trained on a same dataset, but with different initialisations. One can also resort to *Bayesian NNs*, turning the deterministic layer weights into their probabilistic counterparts. This category includes *variational inference* [8] and *Monte Carlo dropout* [9] approaches.

Several authors studied data consistency (DC) for X-ray reconstructions generated by a NN. Some incorporate a minimisation of this DC term inside the optimisation process [10, 11] at the cost of a higher memory-consumption of the training loop. Others consider offline post-processing steps [12–14] but have no control over the location and the variation of the pixels that are updated.

2. METHODS

Dataset To train our deep neural network in a supervised way despite the lack of DBT ground truth artefact-free volumes, we resort to a synthetic dataset made up of phantoms whose texture closely resembles the one of a breast [15]. We use the same methodology as in our previous work [4], but we increase the number of generated phantoms to 288. The volumes are then digitally projected (without noise) and reconstructed iteratively on GPU, before being downsampled to a fifth of the initial resolution for memory concerns. Examples of such phantoms and their associated iterative reconstructions (IR) are displayed in the two left columns of Figure 1. Eventually, we get a paired training dataset from 80 % of the whole database, composed of initial IR $\tilde{\mathbf{x}}_\mathbf{i}$ and corresponding ground truth (GT) phantoms $\mathbf{x}_\mathbf{i}^*$. The remaining 20 % are split equally to form the validation and test sets.

Aleatoric uncertainty In order to estimate the aleatoric uncertainty, we consider that the per-pixel output error $[\boldsymbol{\epsilon}_\mathbf{i}]_j \triangleq [\mathbf{x}_\mathbf{i}^*]_j - [\hat{\mathbf{x}}_\mathbf{i}]_j$ follows a zero-mean Laplace distribution, ensuring a regression that is more robust to outliers than with a Gaussian. Deriving the negative log-likelihood of the above Laplace law and removing the constant terms, we obtain the training loss function of the NN for the image $\mathbf{i}$ with n pixels:

$$\mathcal{L}(\hat{\mathbf{x}}_\mathbf{i}, \hat{\mathbf{b}}_\mathbf{i}) = \frac{1}{n} \sum_{j=1}^{n} \frac{|[\mathbf{x}_\mathbf{i}^*]_j - [\hat{\mathbf{x}}_\mathbf{i}]_j|}{[\hat{\mathbf{b}}_\mathbf{i}]_j} + \log[\hat{\mathbf{b}}_\mathbf{i}]_j,$$

with $\mathbf{x}_\mathbf{i}^*$ the ground truth, $\hat{\mathbf{x}}_\mathbf{i}$ the output reconstruction, and $\hat{\mathbf{b}}_\mathbf{i}$ the pixel-dependent scale parameter. In practice, to predict $\hat{\mathbf{x}}_\mathbf{i}$ and $\hat{\mathbf{b}}_\mathbf{i}$, the head of the NN is split into two branches [6, 16].

Epistemic uncertainty To compute the epistemic uncertainty, we compare the use of a deep ensemble (DE) [7] and a Bayesian network with Monte Carlo dropout (MCDO) [9]. MCDO networks can be loosely seen as an alternative form of ensemble, as dropping neurons during inference yields slightly different NN architectures for each prediction. Both methods are trained, validated and tested on the same datasets. In order to ensure enough diversity in the ensemble, we use the same strategy as in [7]: each network is learnt on the whole training dataset, with the sample images shuffled each time and weights initialised at random. To create the MCDO network, we randomly zero-out entire feature maps before each convolution [17], with a probability $p = 0.1$. The predicted variance (*i.e.,* the squared uncertainty) is then approximated by Monte Carlo integration [7, 16]:

$$[\hat{\boldsymbol{\sigma}}_\mathbf{i}^\mathbf{2}]_j \approx \underbrace{\frac{1}{T}\sum_{t=1}^{T} 2\left([\hat{\mathbf{b}}_\mathbf{i}]_j^t\right)^2}_{\text{Aleatoric term}} + \underbrace{\frac{1}{T}\sum_{t=1}^{T}\left([\hat{\mathbf{x}}_\mathbf{i}]_j^t - \frac{1}{T}\sum_{t=1}^{T}[\hat{\mathbf{x}}_\mathbf{i}]_j^t\right)^2}_{\text{Epistemic term}}$$

with T the number of forward passes (or NNs in the ensemble), $[\hat{\mathbf{b}}_\mathbf{i}]_j^t$ and $[\hat{\mathbf{x}}_\mathbf{i}]_j^t$ respectively the output scale and predicted j-th pixel of the t-th forward pass. In this work, we choose $T = 6$. Remark that in the case of a Laplace with scale b, the variance is given by $2b^2$. The predicted mean (*i.e.,* the reconstruction) is computed as $[\hat{\mathbf{x}}_\mathbf{i}]_j = \frac{1}{T}\sum_{t=1}^{T}[\hat{\mathbf{x}}_\mathbf{i}]_j^t$. According to [18], the biased total variance $\hat{\boldsymbol{\sigma}}_\mathbf{i}^\mathbf{2}$ should reflect the true variance $\mathbb{E}[(\mathbf{x}_\mathbf{i}^* - \hat{\mathbf{x}}_\mathbf{i})^2]$ when the predictions are well optimised.

Data consistency To avoid a large computational intake, we prefer to decouple the data consistency constraint from the training loop [13]. Thereby, we add an offline DC algorithm to our pipeline, taking already reconstructed volumes as input. Common DC algorithms minimise $\|\mathbf{A}\mathbf{x}_\mathbf{i} - \mathbf{y}_\mathbf{i}\|_2$ for a volume $\mathbf{i}$, where $\mathbf{x}_\mathbf{i}$ is initialised with a reconstruction (in our case the output of the MCDO or DE models), and $\mathbf{y}_\mathbf{i}$ are the true projections acquired by the DBT system. However, doing so requires a projection and a back-projection, which reintroduce artefacts in the volume due to the ill-posed nature of the problem. To avoid this effect, we propose to focus the updates onto the uncertain areas only by optimising

$$\arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{A}\mathbf{x}_\mathbf{i} - \mathbf{y}_\mathbf{i}\|_2^2 \quad \text{s.t.} \quad \|(\mathbf{x}_\mathbf{i} - \hat{\mathbf{x}}_\mathbf{i}) \oslash \hat{\boldsymbol{\sigma}}_\mathbf{i}\|_\infty \leq \alpha,$$

with $[\hat{\boldsymbol{\sigma}}_\mathbf{i}]_j = \sqrt{[\hat{\boldsymbol{\sigma}}_\mathbf{i}^\mathbf{2}]_j}$, and α a relaxation parameter which is taken as 1 in this study to keep the same order of magnitude as the pixels of $\hat{\mathbf{x}}_\mathbf{i}$. The symbol $\oslash$ denotes the element-wise division. The second term ensures that the volume stays close to the NN prediction where the uncertainty is low, but allows

for more freedom where it is high. We minimise this optimisation problem using a projected gradient descent, and call this technique *uncDC*. Remark that there is no regularisation as our projections and reconstructions are noiseless.

3. RESULTS

Implementation details The neural network used in this work is a residual 2D U-Net with two output branches (see [4] and Section 2). We employ a 2.5D training strategy: inputs and targets are actually stacks of 7 successive coronal slices, which are thereafter aggregated to create the final reconstruction. The layer weights are optimised using Adam algorithm. We moreover consider that the output branch predicting aleatoric uncertainty actually corresponds to its logarithm. The loss function then becomes $\sum_{ij} |[\mathbf{x}_\mathbf{i}^*]_j - [\hat{\mathbf{x}}_\mathbf{i}]_j| e^{-\log[\hat{\mathbf{b}}_\mathbf{i}]_j} + \log[\hat{\mathbf{b}}_\mathbf{i}]_j$, which results in a better numerical stability.

Evaluation metrics To evaluate the quality of the reconstructed volumes, we choose the well-known Root Mean Squared Error (RMSE) and Structural Similarity Index Measure (SSIM). Besides, to gauge the calibration of the predicted uncertainties, we use the Uncertainty Calibration Error (UCE) [18], which compares the error and the uncertainty, as the latter should reflect the former. All metrics are gathered in Table 1.

Reconstructions and uncertainty maps From Table 1, one can observe that the DE and MCDO performances somewhat surpass the ones of a single 2.5D NN with the same architecture. All three methods improve a lot the quality of the initial reconstruction, yet the advantage of MCDO and DE lie in their ability to compute uncertainty maps.

	RMSE ($\times 10^{-3}$) $\downarrow$	SSIM $\uparrow$	UCE ($\times 10^{-4}$) $\downarrow$
$\tilde{\mathbf{x}}_\mathbf{i}$	6.07 ± 1.12	0.733 ± 0.041	-
NN	3.94 ± 0.91	0.814 ± 0.051	-
DE	**3.52 ± 0.81**	**0.836 ± 0.047**	**4.41 ± 0.75**
MCDO	3.71 ± 0.83	0.825 ± 0.050	6.32 ± 1.25

Table 1: Quantitative metrics averaged on the test set. We recall that $\tilde{\mathbf{x}}_\mathbf{i}$ is the initial iterative reconstruction used as input to our neural network. NN corresponds to the results obtained with a single deterministic 2.5D neural network trained with the same loss function, and hyperparameters.

Figure 1 shows visual results from randomly sampled slices of three different phantoms with varying texture configurations. These images were processed by the best performing model, *i.e.,* the deep ensemble. The true materials' distribution, albeit not perfectly located, is still very well retrieved compared to the input IR. Remark that some areas of the reconstructed volumes are blurred: they correspond to pixels on which the networks in the ensemble could not agree. As anticipated, the uncertainty of these regions is high, reflecting the difficulty the model had to match them to the ground truth. The images clearly show that the inner sections of the material blobs are accurately reconstructed, with minimal uncertainty. However, we observe that the model struggles to retrieve the boundaries between the materials. Vertical borders tend to be thinner on the uncertainty maps than the horizontal ones. This anisotropy is expected: as explained in the introduction, due to the geometry of the imaging system, breast tissues are poorly separated along the vertical axis in the input of the model. As shown in Figure 1, vertical borders are much more visible than horizontal ones. Thereby, recreating the horizontal boundaries at the right height position is a difficult task for the model, as the limit between materials in these regions is unclear. Consequently, an ensemble or MCDO model generating several predictions for a same image may not place the boundaries at the same height each time, resulting in a hazy and spread out uncertainty map.

Accuracy of the uncertainty prediction Calibration can be further analysed with specific plots showing the error as a function of uncertainty. From Figure 2, one can see that for low errors, DE estimates rather well the uncertainty while MCDO tends to overestimate it. For high errors, DE slightly underestimates the uncertainty, while MCDO is well calibrated. Yet, both methods are quite close to the identity line, meaning that the uncertainty estimated by our models can be used as a proxy of the true error. Better curves could be obtained by re-calibrating [18], although applying this method in the case of Laplace distributions would need further investigations.

Data consistency To evaluate the performance of our data consistency block, we compare in Table 2 the relative error between the reprojections of the post-processed volume and those that were acquired with the computer-simulated DBT system. We also monitor the same quantity with relation to the ground truth volume.

	Rel. error $\mathbf{x}_\mathbf{i}^*$ (%) $\downarrow$	Rel. error $\mathbf{y}_\mathbf{i}$ (%) $\downarrow$
DE	7.80 ± 1.85	3.83 ± 1.37
DC - 4 it.	7.69 ± 1.80	1.02 ± 0.29
uncDC - 7 it.	**7.66 ± 1.79**	2.17 ± 1.15
DC - 200 it.	7.89 ± 1.67	0.25 ± 0.18
uncDC - 200 it.	**7.79 ± 1.71**	2.10 ± 1.18

Table 2: Relative errors of the iterated volume and its projections compared respectively to the ground truth $\mathbf{x}_\mathbf{i}^*$ and the acquired projections $\mathbf{y}_\mathbf{i}$. Metrics of the deep ensemble (DE) output volumes are given for comparison.

We perform several iterations of a classic gradient descent

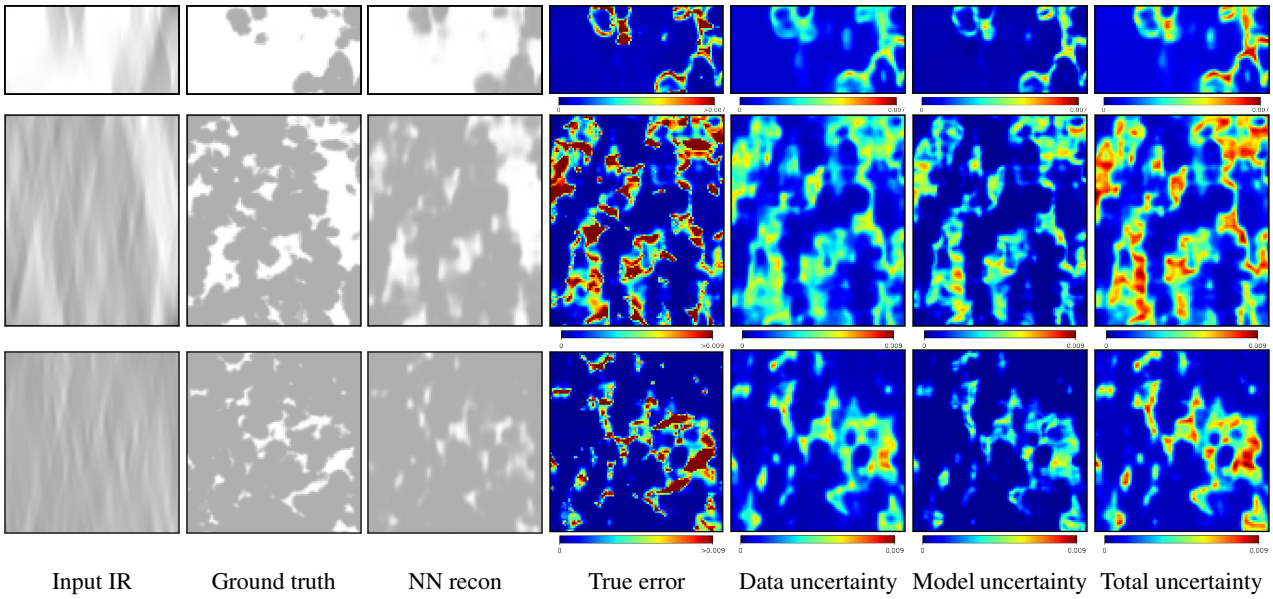
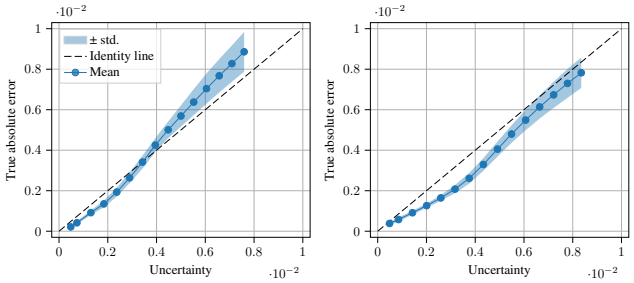


Fig. 1: Random slices taken from the results of three representative phantoms from the test set.



(a) Deep ensemble (b) MCDO

Fig. 2: Calibration plots for the test set. The dashed line corresponds to a perfect calibration.

to minimise $\|\mathbf{A}\mathbf{x}_\mathbf{i} - \mathbf{y}_\mathbf{i}\|_2$ (DC), and do the same for our proposed method (uncDC). After 4 and 7 iterations respectively, the minimum GT error is hit. uncDC yields lower GT error than DC, however, the projection error is around 2 times bigger. We then let the algorithm run until reaching 200 iterations. As expected, the GT error of DC raises because of the reintroduction of artefacts, while the error on the projections keeps being minimised. On the other hand, the GT error of uncDC is still lower than that of the neural network output volume, although it does not stay at the minimum. Overall, our proposed method needs further research, namely regarding a stopping criterion or the benefit of using uncertainty into downstream tasks. We can however conclude that performing very few iterations on the output of the neural network is a good trade-off: it enables to get back some consistency to the projections while staying close to the ground truth despite the reintroduction of artefacts.

4. DISCUSSION AND CONCLUSION

In this paper, we proposed a deep learning pipeline for DBT reconstruction which is able to estimate its own reliability. Yet, our proposed models output accurate reconstructions of coronal planes, despite a huge lack of information in this direction. The two compared methods, deep ensemble and Monte Carlo dropout, enable to compute uncertainties that share the expected geometrical properties of the true error, namely anisotropy along the vertical axis. The algorithms we developed generate a well-calibrated total uncertainty that could be used as a proxy to estimate the true error without access to a ground truth, as it is the case during inference. We thus propose a way to integrate uncertainty into a downstream data consistency task in order to focus the updates onto the uncertain areas. The corresponding results, although preliminary, show that uncertainty can help to get a better estimation of the ground truth reconstruction. However, this method is highly dependent on the quality of the NN reconstruction.

Nevertheless, using the negative log-likelihood as a loss function to train our neural networks can cause faster convergence of the residual error compared to the uncertainty [19]. This is confirmed by the calibration plots in Figure 2 where the points with high error are not fully optimised. To avoid this effect, one could prefer to use a more complex distribution to model the pixel-wise error $[\boldsymbol{\epsilon}_\mathbf{i}]_j$, such as a generalised Gaussian [20], giving the network more control over the outliers by allowing fatter tailed distributions where the acquisition geometry is known to be hard to handle.

5. COMPLIANCE WITH ETHICAL STANDARDS

This is a numerical simulation study for which no ethical approval was required.

6. ACKNOWLEDGEMENTS

This work was partially funded by the French Ministry for Higher Education and Research as part of CIFRE grant No. 2021/1209.

7. REFERENCES

[1] L. T. Niklason *et al.*, "Digital tomosynthesis in breast imaging.," *Radiology*, Nov. 1997.

[2] T. M. Buzug, *Computed Tomography: From Photon Statistics to Modern Cone-Beam CT*. Springer Science & Business Media, May 2008.

[3] I. Reiser and S. Glick, *Tomosynthesis Imaging*. Taylor & Francis, Mar. 2014.

[4] A. Quillent, V. Bismuth, I. Bloch, C. Kervazo, and S. Ladjal, "A deep learning method trained on synthetic data for digital breast tomosynthesis reconstruction," in *Medical Imaging with Deep Learning*, 2023.

[5] J. Gawlikowski *et al.*, "A survey of uncertainty in deep neural networks," *Artificial Intelligence Review*, Jul. 2023, ISSN: 1573-7462.

[6] D. Nix and A. Weigend, "Estimating the mean and variance of the target probability distribution," in *IEEE International Conference on Neural Networks (ICNN'94)*, vol. 1, Jun. 1994, 55–60 vol.1.

[7] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles," in *Advances in Neural Information Processing Systems*, vol. 30, Curran Associates, Inc., 2017.

[8] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra, "Weight Uncertainty in Neural Network," in *32nd International Conference on Machine Learning*, PMLR, Jun. 2015, pp. 1613–1622.

[9] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning," in *33rd International Conference on Machine Learning*, PMLR, Jun. 2016, pp. 1050–1059.

[10] C. Angermann, S. Göppel, and M. Haltmeier, "Uncertainty-Aware Null Space Networks for Data-Consistent Image Reconstruction," *arXiv:2304.06955*, 2023.

[11] J. Teuwen *et al.*, "Deep learning reconstruction of digital breast tomosynthesis images for accurate breast density and patient-specific radiation dose estimation," *Medical Image Analysis*, vol. 71, p. 102 061, Jul. 2021.

[12] Y. Huang *et al.*, "Data Consistent Artifact Reduction for Limited Angle Tomography with Deep Learning Prior," in *Machine Learning for Medical Image Reconstruction*, ser. LNCS, 2019, pp. 101–112.

[13] D. Wu, K. Kim, and Q. Li, "Digital Breast Tomosynthesis Reconstruction with Deep Neural Network for Improved Contrast and In-Depth Resolution," in *17th IEEE International Symposium on Biomedical Imaging (ISBI)*, Apr. 2020, pp. 656–659.

[14] A. Lahiri, G. Maliakal, M. L. Klasky, J. A. Fessler, and S. Ravishankar, "Sparse-View Cone Beam CT Reconstruction Using Data-Consistent Supervised and Adversarial Learning From Scarce Training Data," *IEEE Transactions on Computational Imaging*, vol. 9, pp. 13–28, 2023.

[15] Z. Li *et al.*, "A 3D Mathematical Breast Texture Model with Parameters Automatically Inferred from Clinical Breast CT Images," *IEEE Transactions on Medical Imaging*, vol. 42, no. 4, pp. 1107–1120, 2022.

[16] A. Kendall and Y. Gal, "What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?" In *Advances in Neural Information Processing Systems*, vol. 30, Curran Associates, Inc., 2017.

[17] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler, "Efficient object localization using Convolutional Networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 648–656.

[18] M.-H. Laves, S. Ihler, J. F. Fast, L. A. Kahrs, and T. Ortmaier, "Recalibration of Aleatoric and Epistemic Regression Uncertainty in Medical Imaging," *Machine Learning for Biomedical Imaging*, vol. 1, no. MIDL 2020 special issue, pp. 1–26, Apr. 2021.

[19] M. Seitzer, A. Tavakoli, D. Antic, and G. Martius, "On the Pitfalls of Heteroscedastic Uncertainty Estimation with Probabilistic Neural Networks," in *International Conference on Learning Representations*, Oct. 2021.

[20] U. Upadhyay, Y. Chen, and Z. Akata, "Robustness via Uncertainty-aware Cycle Consistency," in *Advances in Neural Information Processing Systems*, vol. 34, Curran Associates, Inc., 2021, pp. 28 261–28 273.