Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Shape and Attenuation Cues for Defect-Type Classification in X-ray CT: A Controlled Synthetic Study with a 3D CNN

Malavath Rahul
Indian Institute of Technology Roorkee
Mayank Goswami
mayankfph@iitr.ac.in

Indian Institute of Technology Roorkee https://orcid.org/0000-0001-8829-9980

Research Article

Keywords: X-ray computed tomography, defect classification, 3D convolutional neural network, synthetic data, attenuation contrast, ablation study

Posted Date: November 16th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11724976/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

Shape and Attenuation Cues for Defect-Type Classification in X-ray CT: A Controlled Synthetic Study with a 3D CNN

Malavath Rahul (BS PHYSICS),and Mayank Goswami [1, 2, #]
[1] Department of Physics, IIT Roorkee
[2]*Mehta Family School of Data Science and Artificial Intelligence*, IIT Roorkee,
Roorkee, Uttarakhand-247667, India.
[#]mayank.goswami@ph.iitr.ac.in

Abstract—After a candidate defect has been located in an X-ray computed tomography (CT) volume, an inspector must still decide whether it is a pore, a crack or an inclusion, because each type has different acceptance criteria and root causes. Pores and cracks are voids with near-zero attenuation, whereas inclusions have a distinct, non-zero attenuation, so intensity should help where shape alone is ambiguous. We test this in a controlled synthetic study: 7,500 exactly labelled 3D patches (2,500 per class) generated from 200 simulated scans with partial-volume blur, shading and Poisson–Gaussian noise at three SNR levels, split by scan into training, validation and test sets. A compact 3D CNN (0.88 M parameters) reaches 0.990 test accuracy (0.990 ± 0.002 over five training seeds; 95 % CI of the headline run 0.984–0.996). An identical network given only binary masks reaches 0.824 ± 0.015, a drop of 16.6 ± 1.6 points. In the headline run (drop 17.5 points; McNemar exact $p \approx 10^{-50}$) the loss is confined to pore–inclusion confusion, cracks being recognised perfectly, and comes almost entirely from spherical inclusions, whose masks resemble spherical pores. Random Forest and SVM classifiers on shape and intensity features perform similarly (0.988–0.997), so the CNN is competitive rather than superior. All 11 errors of the headline run are either dark inclusions with contrast close to that of a void or pores and cracks of 59–69 voxels, and accuracy did not degrade from SNR 40 to 10. Because the data are synthetic, the results are an upper bound for real scans.

Keywords—X-ray computed tomography; defect classification; 3D convolutional neural network; synthetic data; attenuation contrast; ablation study

1 Introduction

X-ray computed tomography (CT) is widely used to inspect the interior of castings, welds and additively manufactured parts without destroying them. Recent reviews describe a shift from hand-designed image-processing pipelines towards deep learning for detection, segmentation and classification [1]. Most work addresses detection or segmentation; here we study the next step, deciding the type of a defect that has already been located.

The distinction matters because pores, cracks and non-metallic inclusions are judged against different acceptance criteria and point to different process causes. They differ in shape, but also in their attenuation signature. Voids absorb X-rays like air, so a pore and a crack cannot be separated by an intensity threshold and need shape cues [2], whereas an inclusion of foreign material has an attenuation that is higher or lower than that of the matrix. A purely geometric classifier cannot exploit this cue and should therefore confuse pores with inclusions of similar shape.

This paper quantifies that expectation. Defect type and attenuation contrast are only unambiguously known when they are set during generation, so we use exactly labelled synthetic data, an approach that has also been used to train defect detectors later tested on real parts [3]. Our contributions are:

- a reproducible generator of volume-matched pore, crack and inclusion patches with a scanner model (blur, shading, Poisson–Gaussian noise) and per-instance logged contrast, built so that defect size cannot serve as a shortcut;
- a controlled ablation that isolates the contribution of attenuation from geometry, with scan-level confidence intervals and McNemar's test;
- a fair comparison with classical shape-plus-intensity classifiers, using both exact and automatically thresholded defect masks; and
- an error analysis that ties every misclassification to a physical cause.

2 Related Work

CT defect analysis. Noise, low contrast-to-noise ratio and reconstruction artefacts hinder threshold-based segmentation, which motivated deeply supervised 3D networks for pore segmentation in additively manufactured

samples [4]; deep learning has also been used to detect and grade defect types in high-pressure aluminium castings from X-ray images [5]. Partial-volume blur, which can be reduced but not removed [6], makes the measured contrast of small defects weaker than their nominal contrast, so we model it explicitly.

Classifiers and statistics. Feature-based Random Forests [7] and support vector machines [8] are interpretable but depend on a prior segmentation. 3D CNNs learn from voxel grids directly: VoxNet applied 3D convolutions to 32×32×32 occupancy grids [9], and the 3D U-Net extended them to volumetric segmentation [10]. We follow this practice with a 32^3 input, batch normalisation [11] and the Adam optimiser [12]. For comparing classifiers on one test set, McNemar's test [13] was found by Dietterich to have acceptable type-I error when an algorithm can be run only once, although it ignores training variability [14]. In this limited review we did not find a controlled quantification of what attenuation adds beyond geometry for 3D defect-type classification; this study addresses that question.

3 Synthetic Data

3.1 Generators and volume matching

Three generators produce the classes (Fig. 1 and Fig. 2). Pores are spherical, ellipsoidal (axis ratios up to 2.2) or irregular (ratios up to 1.6 plus a smooth random surface perturbation). Cracks are thin sheets, 1.2–2.4 voxels thick, with an elliptical, ragged outline, either planar or curved (quadratic bending up to ±3 voxels). Inclusions are spheres or convex polyhedra (6–14 faces, elongation up to 1.5). Voids take 0–5 % of the matrix attenuation, whereas an inclusion takes $m(1 + c)$, with contrast $c = \pm U(0.05, 0.85)$ of random sign and m the matrix level. Each generator ranks the voxels of a 40^3 cell by a shape-specific score and keeps exactly the V best-scoring voxels, so the defect volume is hit exactly for every shape.

The volume V is drawn log-uniformly from 60–600 voxels for all classes. Without this, thin cracks could be separated from pores by the fraction of the patch they fill, a shortcut unrelated to morphology. Two-sample Kolmogorov–Smirnov tests found no difference between the realised volumes ($p = 0.15$, 0.94 and 0.25 for pore–crack, pore–inclusion and crack–inclusion; medians 190, 186 and 189 voxels).

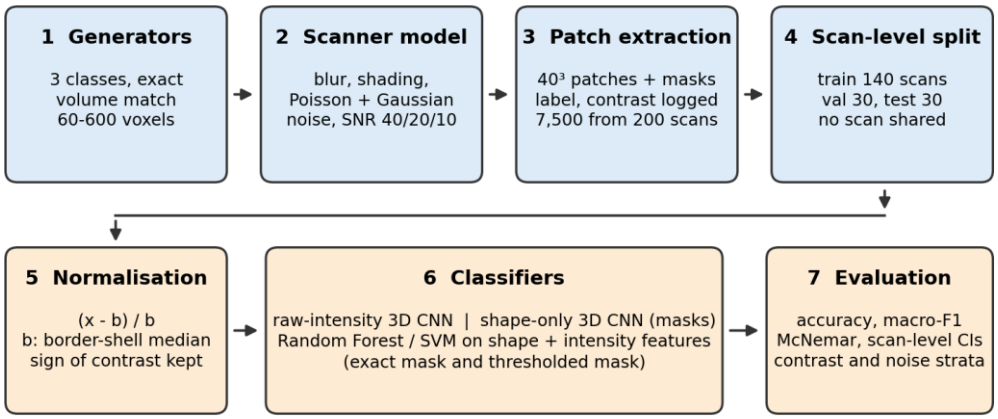


Fig. 1. Overview of the study: synthetic generation, scanner model, patch extraction and scan-level split, followed by normalisation, the classifiers and the evaluation.

3.2 Scanner model and scan-level split

Each of 200 simulated scans is a 176^3 grid of 4×4×4 cells holding 36–39 defects, one per cell. Per scan we draw the matrix level $m \in [0.8, 1.2]$, a Gaussian point-spread function with $\sigma \in [0.6, 0.9]$ voxels (the partial-volume effect), a linear shading gradient of up to 3 %, and one of three noise tiers. Noise combines Poisson and Gaussian components, each contributing half of the variance at the matrix level and lightly smoothed to mimic reconstructed CT noise; its standard deviation is m/SNR with SNR = 40, 20 or 10. Noise is added before patches are cut. A 40^3 patch centred on each defect is stored with its exact mask, and class, sub-type, scan ID, SNR, signed contrast and volume are logged.

The split is decided per scan: 140 scans for training, 30 for validation and 30 for testing, giving exactly 5,250 / 1,125 / 1,125 patches (1,750 / 375 / 375 per class) with noise tiers spread evenly across splits. An assertion confirmed that

no scan appears in two splits, and no patch contains voxels of a neighbouring defect. The test set holds 54 subtle ($|c| < 0.2$), 142 moderate (0.2–0.5) and 179 obvious (≥ 0.5) inclusions.

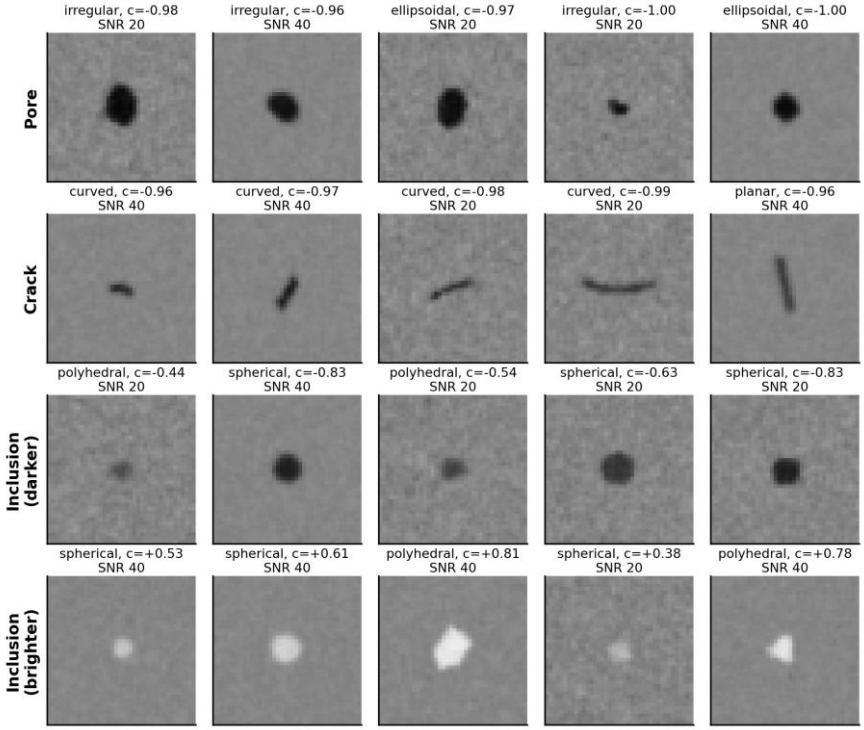


Fig. 2. Example test patches (central slice; matrix = 0 grey, void = black). Rows: pores, cracks, darker and brighter inclusions. Titles give the sub-type, contrast c and SNR.

3.3 Preprocessing

Per-patch min–max scaling would erase whether an inclusion is brighter or darker than the matrix, the very cue under test. We instead normalise $x' = (x - b)/b$, where b is the median of the outer 4−voxel border shell, which contains matrix only. The matrix then maps to 0, a void to about −1 and an inclusion to its relative contrast (Fig. 3): on the test set the mean normalised intensity inside inclusions correlates with the generating contrast at $r = 0.996$. Training uses random 32^3 crops from the 40^3 patch, random axis permutations and flips (48 orientations) and, for the raw-intensity model only, a small contrast-preserving jitter (gain 0.95–1.05, offset ±0.02, extra noise $\sigma \leq 0.02$). The shape-only model receives the exact binary mask with the same crops and orientations.

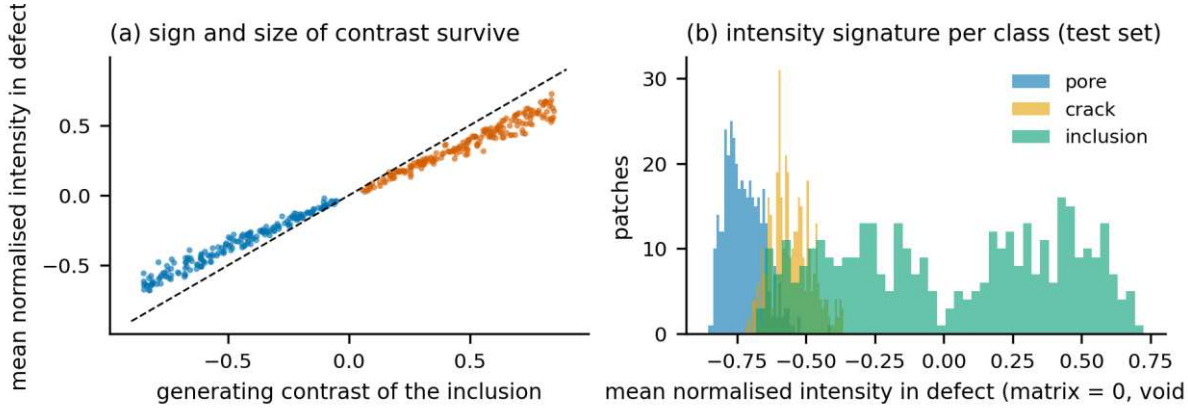


Fig. 3. Normalisation preserves the attenuation signature. (a) Mean normalised intensity inside each inclusion versus its generating contrast; the small shortfall is partial-volume blur. (b) Intensity signature of the three classes on the test set.

4 Methods

4.1 Models

The 3D CNN (0.88 M parameters) has four blocks of two 3×3×3 convolutions with batch normalisation and ReLU, each followed by max-pooling, with 16, 32, 64 and 128 channels; global average pooling and a 128→32→3 head with dropout 0.3 give the class scores. Training used cross-entropy (no class weights, as the classes are balanced), Adam with learning rate 10^{-3}, batch size 64, a plateau scheduler and early stopping on validation macro-F1 (patience 12, at most 60 epochs), mixed precision and seed 42. The raw-intensity model stopped after 46 epochs and the shape-only model used all 60 (best validation macro-F1 0.996 and 0.824); an epoch took about 5 s on a Tesla T4. To measure training variability, both networks were also retrained with four further seeds (1–4) under identical settings. The shape-only ablation is identical in architecture, seed and schedule; only the input differs. Because it receives the exact mask it represents the best case for geometry, so the measured contribution of intensity is conservative.

4.2 Classical baselines

Eight shape descriptors (volume, sphericity, solidity, bounding-box fill, three principal-axis ratios, maximum extent) and seven intensity statistics (mean, standard deviation and skewness inside the defect; mean of a ring around it; mean, standard deviation and skewness of the central patch) feed a Random Forest [7] or an RBF-kernel SVM [8], tuned on validation macro-F1 only. We report four variants: Random Forest and SVM on all features with the exact mask (an oracle that never has to find the defect), a Random Forest on shape features only, and a Random Forest on all features computed from a thresholded mask (Gaussian smoothing, a threshold at max(4 × robust noise, 0.05), the component nearest the centre, and edge refinement at half peak contrast; Dice 0.94 for pores, 0.77 for cracks and 0.59 / 0.82 / 0.92 for subtle / moderate / obvious inclusions). The last variant is the fairer counterpart to a CNN that works on the raw patch.

4.3 Evaluation

We report accuracy, macro-F1, per-class precision and recall and confusion matrices on the held-out test set, overall and by contrast stratum and noise tier. Confidence intervals come from 500 bootstrap resamples of whole scans [15], because patches from one scan are not independent; proportions in small strata use Wilson intervals [16]. Models are compared with McNemar's test on the same 1,125 test patches (exact binomial p). Implementation used PyTorch [17], scikit-learn [18], scikit-image [19] and SciPy [20].

5 Results

5.1 Overall performance

The raw-intensity CNN of the headline run (seed 42) reaches 0.990 accuracy and macro-F1 on the 1,125 test patches, with 11 errors (Table 1, Fig. 4). Over five training seeds (Table 2) it reaches 0.990 ± 0.002 accuracy with 9–14 errors, and the shape-only CNN 0.824 ± 0.015, a drop of 16.6 ± 1.6 points (range 14.5–18.1); the 17.5-point drop of the headline run has a scan-level bootstrap 95 % CI of 15.1 to 19.6 points.

Table 1. Test-set performance (n = 1,125; 95 % CIs from resampling scans). CNN rows are the headline run (seed 42). RF = Random Forest; "exact" and "thresholded" refer to the defect mask used for the features.

Model	Accuracy [95 % CI]	Macro-F1	Recall pore	Recall crack	Recall incl.
3D CNN, raw intensity	0.990 [0.984, 0.996]	0.990	0.995	0.992	0.984
3D CNN, shape-only	0.815 [0.796, 0.838]	0.810	0.883	1.000	0.563
RF, shape + intensity (exact)	0.991 [0.986, 0.997]	0.991	0.989	0.997	0.987
SVM, shape + intensity (exact)	0.997 [0.994, 1.000]	0.997	0.995	0.997	1.000
RF, shape only (exact)	0.836 [0.812, 0.860]	0.834	0.651	0.997	0.861
RF, shape + intensity (thresholded)	0.988 [0.981, 0.996]	0.988	0.976	0.995	0.995

Table 2. Training variability: the headline run (seed 42) and four repeats with identical settings. Drop = raw-intensity minus shape-only accuracy, in percentage points. Last row: mean ± sample standard deviation over the five runs.

Seed	Raw accuracy	Raw errors	Shape-only accuracy	Drop (points)	Shape-only inclusion recall
42 (headline)	0.990	11	0.815	17.5	0.563
1	0.992	9	0.811	18.1	0.813
2	0.988	13	0.814	17.4	0.651
3	0.988	14	0.843	14.5	0.811
4	0.991	10	0.837	15.4	0.683
Mean ± std	0.990 ± 0.002	11.4 ± 2.1	0.824 ± 0.015	16.6 ± 1.6	0.704 ± 0.108

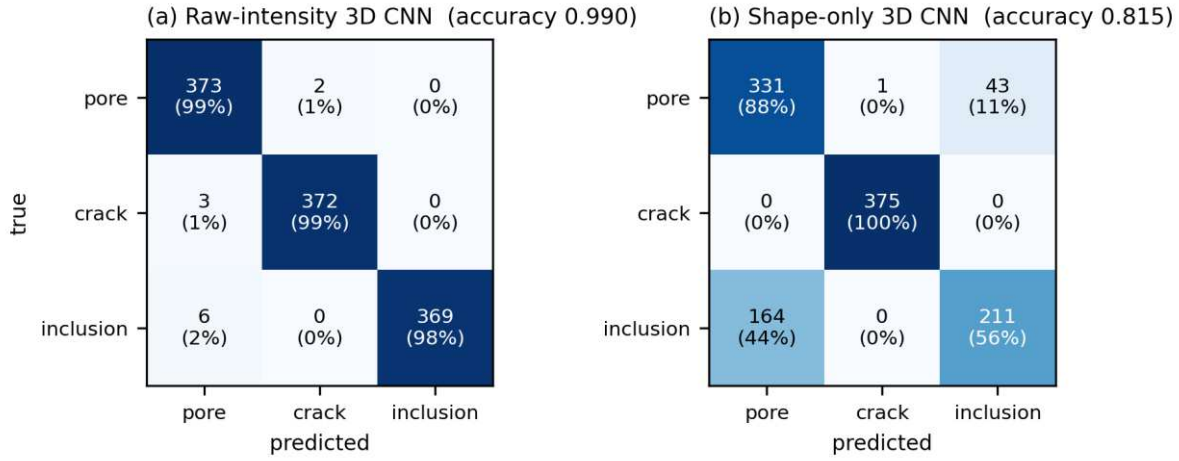


Fig. 4. Confusion matrices on the test set for the headline run (seed 42; rows: true class). (a) Raw-intensity CNN. (b) Shape-only CNN: all errors are pore–inclusion confusion.

5.2 What attenuation contributes

In the headline run, removing intensity cost 17.5 accuracy points (0.990 → 0.815), and 16.6 ± 1.6 points over the five seeds. McNemar's test found 205 patches that only the raw-intensity model classified correctly against 8 that only the shape-only model did (χ^2 = 180.4, exact $p = 1.5 \times 10^{-50}$). The loss is confined to pores and inclusions (Fig. 4b): the shape-only network classifies all 375 cracks correctly but calls 164 of 375 inclusions pores and 43 of 375 pores inclusions. The sub-type breakdown (Fig. 5) explains why: it recognises 94 % of polyhedral inclusions from their facets but only 17 % of spherical inclusions, whose masks are indistinguishable from spherical pores; these account for 153 of its 164 inclusion errors. Across the five seeds the shape-only accuracy is stable (0.811–0.843), but the balance of its errors is not: its inclusion recall ranges from 0.56 to 0.81 (Table 2), which is consistent with the mask of a sphere not telling the network whether it is a pore or an inclusion, so that its choice for such masks varies from run to run. The sub-type figures above therefore describe the headline run, and the mechanism rather than the exact split is the robust result. A different shape-only model, the Random Forest on exact-mask shape features, reaches 0.836, which supports 0.82–0.84 as the ceiling for geometry on this data.

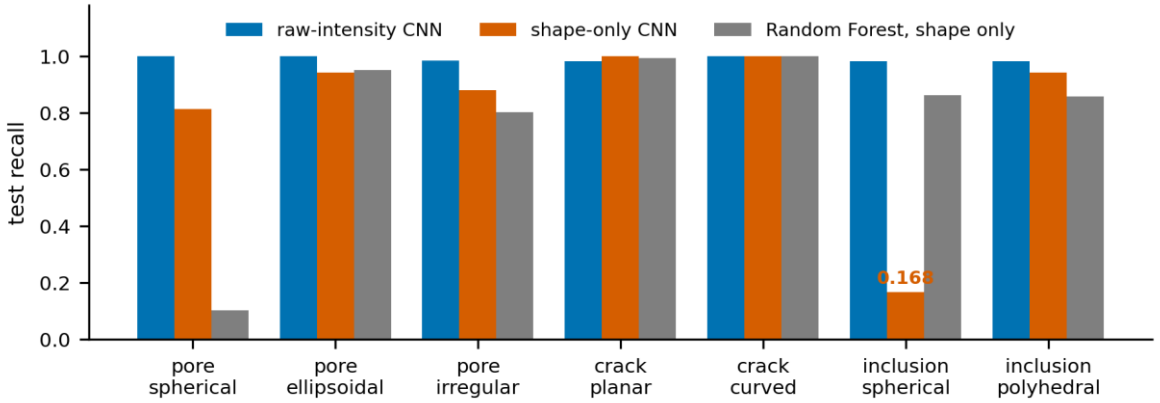


Fig. 5. Test recall per defect sub-type. The shape-only CNN fails almost exclusively on spherical inclusions (recall 0.168), which look like spherical pores.

5.3 CNN versus classical baselines

With exact masks the Random Forest (0.991) and SVM (0.997) match the CNN, and with a thresholded mask the Random Forest reaches 0.988 (Table 1). Paired tests (Table 3) show no significant difference between the CNN and either Random Forest; the SVM is nominally better (exact $p = 0.039$, from 12 discordant patches), which does not survive a Bonferroni correction for the four comparisons ($\alpha = 0.0125$). The Random Forest relies mainly on the mean and standard deviation of intensity inside the defect; removing the intensity features lowers it from 0.991 to 0.836, so attenuation adds about 15.5 points for classical models as well.

Table 3. McNemar's test on paired test predictions (CNNs: headline run, seed 42).

Comparison	Only first correct	Only second correct	Exact p
Raw CNN vs shape-only CNN	205	8	1.5×10^{-50}
Raw CNN vs RF (exact mask)	6	7	1.00
Raw CNN vs SVM (exact mask)	2	10	0.039
Raw CNN vs RF (thresholded mask)	8	6	0.79
RF with vs without intensity features	178	4	1.5×10^{-47}

5.4 Contrast and noise

Subtle inclusions were not the hard case: the CNN classified all 54 correctly (Fig. 6a), as did every intensity-aware model, because pores and cracks are always strongly dark and a faint blob is an inclusion by elimination. The only inclusion errors lie in the obvious stratum, and all six fall between −0.85 and −0.75, i.e. dark inclusions whose attenuation is close to that of a void: in the −0.85 to −0.70 bin (n = 36) inclusion recall is 0.833 (Wilson 95 % interval 0.68–0.92) and it is 1.000 elsewhere (Fig. 6b). In the headline run the shape-only CNN is flat, with 39–48 % of inclusions called pores at every contrast, because it cannot see contrast. Accuracy did not fall from SNR 40 to SNR 10 for any model (Table 4); the raw CNN scored 0.987, 1.000 and 0.984, differences of a few patches within sampling variation, and it classified every subtle inclusion correctly at each tier.

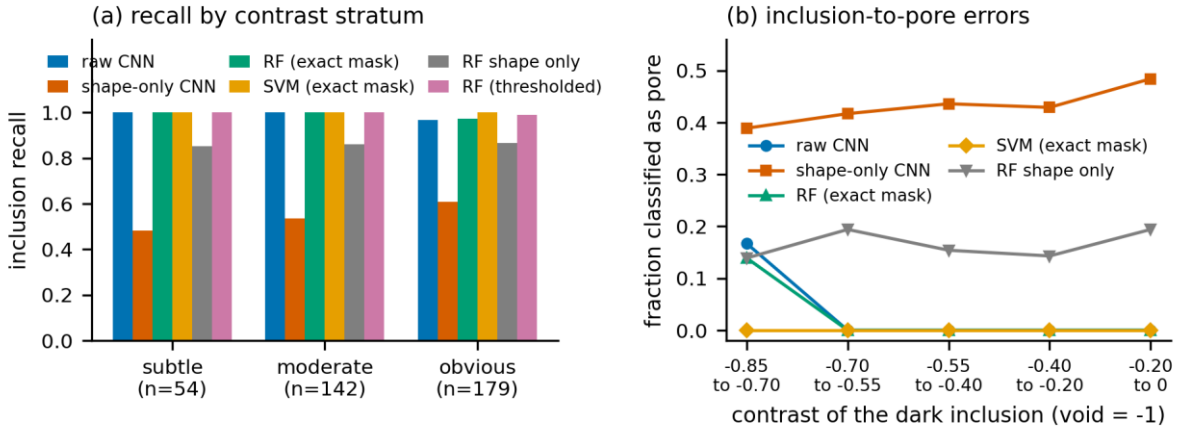


Fig. 6. Contrast analysis. (a) Inclusion recall by contrast stratum (subtle $|c| < 0.2$, moderate 0.2–0.5, obvious ≥ 0.5). (b) Fraction of dark inclusions classified as pore by contrast bin; only the darkest bin, closest to a void, produces errors for the intensity-aware models.

Table 4. Test accuracy by noise tier (about 375 patches per tier).

Model	High (SNR 40)	Medium (SNR 20)	Low (SNR 10)
3D CNN, raw intensity	0.987	1.000	0.984
3D CNN, shape-only	0.816	0.804	0.825
RF, shape + intensity (exact)	0.987	1.000	0.987
SVM, shape + intensity (exact)	1.000	1.000	0.992
RF, shape only (exact)	0.816	0.878	0.815
RF, shape + intensity (thresholded)	0.987	1.000	0.978

5.5 Error analysis

The 11 errors of the raw-intensity CNN fall into two physical groups (Fig. 7). Six are dark inclusions with contrast −0.76 to −0.85, whose internal level is 15–24 % of the matrix against 0–5 % for a void; four of them occur at the highest SNR, so they are limited by contrast and blur, not by noise. The other five are pore–crack confusions at 59–69 voxels, the lower end of the size range: a planar crack of about 65 voxels is a disc of roughly 3.6 voxels radius and 1.5 voxels thickness, which the point-spread function turns into a small blob. None of the largest third of dark inclusions (contrast ≤ -0.4) was misclassified.

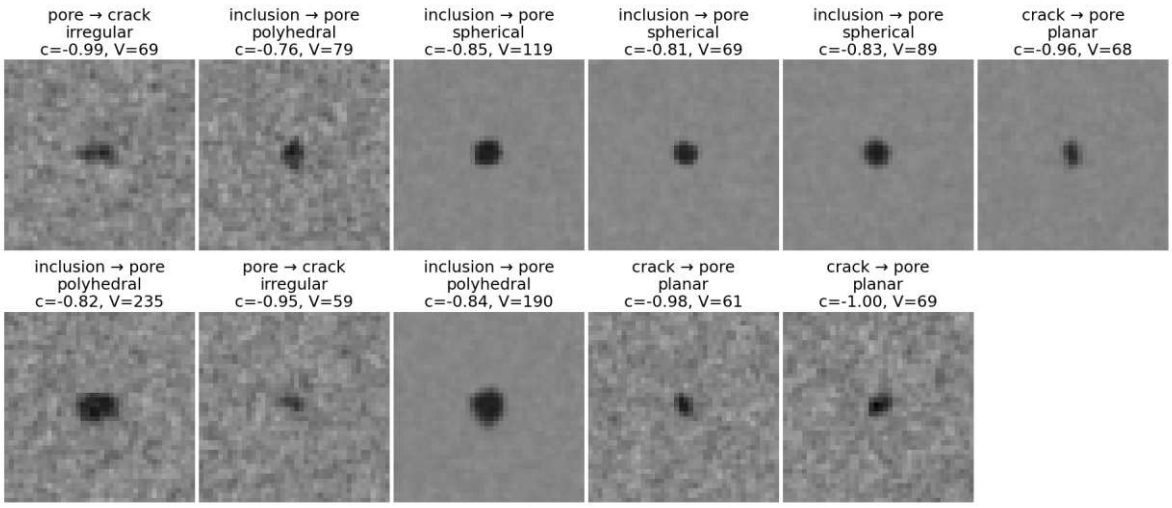


Fig. 7. All 11 test errors of the raw-intensity CNN (central slice), with true → predicted class, sub-type, contrast c and volume V in voxels.

6 Discussion

The accuracy of 0.990 exceeds what we expected, so it deserves scrutiny. We found no sign of leakage: the split is by scan with verified zero overlap, and test accuracy matches validation. The more plausible explanation is that the task is well posed and easy by construction, with mostly well-separated attenuation signatures, defects of at least 60 voxels, exact labels and none of the complications of real scanners (beam hardening, ring and streak artefacts, structured backgrounds, ambiguous ground truth). The numbers are therefore an upper bound for real data.

Intensity matters, but only where shapes overlap: cracks are recognised from geometry alone, whereas pores and inclusions are separated by attenuation wherever their shapes are alike. The size of the ablation drop (16.6 ± 1.6 points over five seeds) is partly a property of our generator, since half of all inclusions are spheres and real inclusions are rarely perfect spheres; the transferable result is the mechanism (Fig. 5), not the number 17.5. The classical baselines are competitive because the discriminating information is low-dimensional, the mean attenuation inside the defect plus a few shape ratios. The potential advantages of the CNN, working on raw patches without any segmentation and exploiting texture, are not tested by this data. Segmentation is the weak point of the classical pipeline: thresholding recovers subtle inclusions poorly (Dice 0.59). Faint inclusions are hard to detect but not hard to classify once a candidate is given, and a real inspection would also have to reject artefacts that mimic them.

6.1 Limitations and future work

(i) The data are synthetic; shapes are idealised, voids are near-zero by construction, and the contrast and size ranges are our choice, so transfer to real scans is untested. (ii) The headline comparison and McNemar's test use one trained model per input type (seed 42), and McNemar's test ignores training variability [14]; the four extra seeds show that the CNN accuracies are stable (Table 2), but the classical baselines were not repeated and the error analysis describes the headline run. (iii) Patches from one scan share noise and blur settings but McNemar's test treats them as independent; confidence intervals account for this by resampling scans. (iv) Four pairwise tests were run, and the near-void contrast bin was defined after inspecting the errors; with $n = 36$ its interval is wide, and the test set holds only 54 subtle inclusions. (v) The shape-only models and one baseline family use exact masks. (vi) Each patch holds one defect of three classes, with no defect-free or artefact class. Future work should validate on real CT scans or use domain-randomised training with a sim-to-real test, add scanner artefacts, use harder noise and smaller defects to find a breaking point, and build an end-to-end detector that finds and classifies defects without a prior mask.

7 Conclusion

A compact 3D CNN classifies synthetic CT defect patches as pore, crack or inclusion with 0.990 test accuracy (0.990 ± 0.002 over five training seeds). Attenuation information is decisive wherever shapes overlap: the same network given only masks falls to 0.824 ± 0.015 over five seeds (0.815 in the headline run, where the loss is entirely pore–inclusion confusion and cracks are still recognised from geometry alone). Hand-crafted shape and intensity features perform about as well on this task, so the CNN's value lies in working on raw patches rather than in higher accuracy. The remaining errors are dark inclusions near void contrast and the smallest pores and cracks, limited by resolution and partial-volume blur rather than noise. Since all data are synthetic, the conclusions about mechanisms are more reliable than the absolute accuracies.

Data and Code Availability

The generator, training and evaluation code are provided as four Google Colab notebooks (data generation; preprocessing and CNN training; baselines and experiments; five-seed repeats), submitted with this preprint and available from the author on request. All data are synthetic and can be regenerated exactly with seed 42.

Acknowledgements

X-Ray CT SMILE Facility IITR

Video Resource: https://youtu.be/A1HB4pvVIQg

References

[1]X. Wen, S. Chen, K. Song, H. Yu, X. Li, and L. Zhong, "X-ray techniques for defect detection in industrial components and materials: A review," Comput. Mater. Contin., vol. 85, no. 3, pp. 4173–4201, 2025, doi: 10.32604/cmc.2025.070906.

[2]E. T. V. Bezerra, K. S. Augusto, and S. Paciornik, "Discrimination of pores and cracks in iron ore pellets using deep learning neural networks," REM – Int. Eng. J., vol. 73, no. 2, pp. 197–203, 2020, doi: 10.1590/0370-44672019730119.

[3]M. Bimrose et al., "Detecting and classifying hidden defects in additively manufactured parts using deep learning and X-ray computed tomography," J. Intell. Manuf., 2024, doi: 10.1007/s10845-024-02416-0.

[4]D. Iuso, S. Chatterjee, R. Heylen, S. Cornelissen, J. De Beenhouwer, and J. Sijbers, “Evaluation of deeply supervised neural networks for 3D pore segmentation in additive manufacturing,” Proc. SPIE 12242, 122421K, 2022, doi: 10.1117/12.2633318.

[5]İ. E. Parlak and E. Emel, “Deep learning-based detection of internal defect types and their grades in high-pressure aluminum castings,” Measurement, 2025, doi: 10.1016/j.measurement.2024.116119.

[6]M. Kato, M. Takahashi, S. Kawasaki, et al., “Evaluation of porosity and its variation in porous materials using microfocus X-ray computed tomography considering the partial volume effect,” Mater. Trans., vol. 54, no. 9, pp. 1678–1685, 2013, doi: 10.2320/matertrans.M-M2013813.

[7]L. Breiman, “Random forests,” Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001.

[8]C. Cortes and V. Vapnik, “Support-vector networks,” Mach. Learn., vol. 20, no. 3, pp. 273–297, 1995.

[9]D. Maturana and S. Scherer, “VoxNet: A 3D convolutional neural network for real-time object recognition,” in Proc. IEEE/RSJ IROS, 2015, pp. 922–928, doi: 10.1109/IROS.2015.7353481.

[10]Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, “3D U-Net: Learning dense volumetric segmentation from sparse annotation,” in MICCAI 2016, LNCS 9901, pp. 424–432.

[11]S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” in Proc. ICML, 2015, pp. 448–456.

[12]D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in Proc. ICLR, 2015, arXiv:1412.6980.

[13]Q. McNemar, “Note on the sampling error of the difference between correlated proportions or percentages,” Psychometrika, vol. 12, no. 2, pp. 153–157, 1947.

[14]T. G. Dietterich, “Approximate statistical tests for comparing supervised classification learning algorithms,” Neural Comput., vol. 10, no. 7, pp. 1895–1923, 1998, doi: 10.1162/089976698300017197.

[15]B. Efron and R. J. Tibshirani, An Introduction to the Bootstrap. New York, NY, USA: Chapman & Hall, 1993.

[16]E. B. Wilson, “Probable inference, the law of succession, and statistical inference,” J. Amer. Statist. Assoc., vol. 22, no. 158, pp. 209–212, 1927.

[17]A. Paszke et al., “PyTorch: An imperative style, high-performance deep learning library,” in Adv. Neural Inf. Process. Syst. 32, 2019.

[18]F. Pedregosa et al., “Scikit-learn: Machine learning in Python,” J. Mach. Learn. Res., vol. 12, pp. 2825–2830, 2011.

[19]S. van der Walt et al., “scikit-image: Image processing in Python,” PeerJ, vol. 2, p. e453, 2014.

[20]P. Virtanen et al., “SciPy 1.0: Fundamental algorithms for scientific computing in Python,” Nat. Methods, vol. 17, pp. 261–272, 2020.