

RESEARCH ARTICLE

YOLOv8-Based Automatic Lesion Detection in Breast Ultrasound Images: A Reproducible Evaluation on a Small and Imbalanced Dataset

Dilara OĞUZ^a, Sevdanur GENÇ^{b,*}, Özlem Şevval SÖNMEZ^c,

^a Department of Computer Engineering, Bursa Technical University, 16350, Bursa, Turkey

^b Department of Computer Engineering, Balıkesir University, 10145, Balıkesir, Türkiye

^c Department of Medicine, Balıkesir University, 10145, Balıkesir, Türkiye

* Corresponding author: sevdanur.genc@balikesir.edu.tr · ORCID: 0000-0003-4774-9265

ARTICLE INFO

Article history

Received 23 June 2026

Revised 28 June 2026

Accepted 28 June 2026

Available online 29 June 2026

Keywords

Breast ultrasound

Object detection

YOLOv8

Computer-aided diagnosis

Class imbalance

Deep learning

Abstract

Early detection is among the strongest determinants of survival in breast cancer, and ultrasound imaging is a routine component of the diagnostic pathway, particularly for patients with dense breast tissue. This study applies YOLOv8n, a lightweight single-stage object detector, to the task of localising lesions in breast ultrasound images and separating benign from malignant findings. Pixel-level lesion masks from the 780 images of the publicly available Breast Ultrasound Images (BUSI) dataset were converted into axis-aligned bounding boxes by external contour analysis, while lesion-free images were retained as unlabelled negative examples. Images were partitioned class-wise into training, validation and test subsets in a 70/15/15 ratio, and class-specific offline augmentation was applied to the training subset only, which brought the numbers of benign and malignant training instances to approximately the same level (614 versus 592). The model was trained for 50 epochs at an input resolution of 640×640. On the test subset, which was used neither for training nor for model selection, the mean average precision at an intersection-over-union threshold of 0.50 (mAP@50) was 0.737 and mAP@50–95 was 0.506. Class-wise mAP@50 was 0.817 for benign and 0.656 for malignant lesions. The findings indicate that a lightweight single-stage detector can reach a reasonable localisation performance on a small ultrasound dataset, but that performance on the malignant class — the clinically decisive one — is not yet adequate for decision-support use. The contribution of this work is not a claim of high accuracy but a reproducible account of every step from data preparation to evaluation, together with an explicit statement of the limitations of the approach.

1. Introduction

Breast cancer is the most frequently diagnosed cancer among women and remains one of the leading causes of cancer death worldwide [1]. Because prognosis depends heavily on the stage at diagnosis, early and accurate identification of lesions is clinically decisive. Mammography forms the basis of population screening programmes, but its sensitivity decreases in patients with dense breast tissue and in younger women; in these situations ultrasound is widely used as a complementary examination. Ultrasound involves no ionising radiation, is portable and is comparatively inexpensive. On the other hand, ultrasound images suffer from speckle noise, low contrast and operator-dependent variability, which makes their interpretation sensitive to the experience of the reader.

Deep learning has become a standard tool in medical image analysis over the past decade [2]. Work on breast ultrasound has concentrated largely on two tasks: classification of the image as a whole and pixel-level segmentation of the lesion region. **Object detection**, which localises the lesion with a bounding box and assigns it a benign or malignant label in the same pass, has received comparatively less attention. Yet a detection output is close to the form required by a clinical workflow: it tells the reader where the lesion is, which class it is assigned to, and how confident the model is. Detection models can also be trained with a lighter annotation burden than segmentation models, which require pixel-level masks.

The YOLO (You Only Look Once) family, the most widely used representative of single-stage detection, treats detection as a regression problem solved in a single forward pass [3] and in its later versions has established a strong balance between accuracy and inference speed [4]. YOLOv8 is a current implementation that uses an anchor-free detection head with decoupled classification and regression branches [5]. The purpose of this study is not to propose a new architecture but to document, **transparently and reproducibly**, the realistic performance limits of YOLOv8 on a small and markedly imbalanced public ultrasound dataset. Reported results on the BUSI dataset span a wide range in the literature, yet the partitioning strategy, the subset to which augmentation was applied, and whether the reported numbers come from a validation split or from a genuinely held-out test split are often left unstated. This ambiguity weakens the comparability of published results.

1.1. Contributions

- (i) A mask-to-box conversion procedure with explicitly stated intensity and minimum-area criteria is defined, producing axis-aligned bounding boxes in YOLO format; lesion-free images are retained as negative examples through empty label files.
- (ii) Class-specific offline augmentation is used to mitigate class imbalance and is applied to the training subset only, so that augmented copies cannot leak into the validation or test subsets.
- (iii) Results are reported class-wise and together with inference timings for both the validation subset and a test subset that was never used during training or model selection; a qualitative error analysis of a false-negative case is also provided.

1.2. Notation and organisation

Throughout, an image is denoted by I and a bounding box by $b = (x, y, w, h)$, where (x, y) is the centre of the box and (w, h) its width and height, all normalised to $[0, 1]$ with respect to the image dimensions. Section 2 reviews related work, Section 3 states the problem and the evaluation measures, Section 4 develops the method, Section 5 reports the numerical experiments, and Section 6 discusses limitations and conclusions.

2. Related Work

Deep learning in breast ultrasound. A substantial part of the work on breast ultrasound analysis uses segmentation methods based on encoder–decoder architectures, of which U-Net is the reference model [9]. Segmentation is valuable because it allows quantitative measurement of lesion boundaries; however, pixel-level annotation is costly and converting the output into a clinical decision requires an additional classification step.

Object detection. Detection methods divide broadly into two-stage and single-stage approaches. Two-stage methods such as Faster R-CNN first generate region proposals and then classify them, generally achieving high accuracy [8]. Single-stage methods, represented by the YOLO family, predict location and class in a single network pass and therefore reach far higher inference rates [3], [4]. YOLOv8 employs an anchor-free head that adopts the distribution focal loss formulation for box regression [14]; this design models the uncertainty of box boundaries probabilistically, which is useful for small objects with variable shape.

Dataset. BUSI is one of the most widely used public datasets in breast ultrasound. It contains 780 images in three classes — normal, benign and malignant — together with the corresponding binary lesion masks [6]. Duplicate images and some inconsistent annotations have subsequently been reported in the dataset [7], which means that performance figures obtained on it should be interpreted with care.

Class imbalance. In imbalanced datasets the performance on the minority class is systematically lower. The principal remedies are reweighting of the loss function [15] and balancing the number of examples through data augmentation [13]. The present work adopts the second approach because its implementation is transparent and repeatable; the geometric and photometric transformations are performed with the Albumentations library [12].

3. Problem Formulation

3.1. Dataset

The study uses the publicly available BUSI dataset, which comprises 780 breast ultrasound images acquired from 600 female patients [6]. The images are grouped into benign (437), malignant (210) and normal (133) classes, and a binary lesion mask delineated by an expert is supplied for every image containing a lesion. Figure 1 shows the class distribution of the raw dataset: the benign class contains more than twice as many examples as the malignant class.

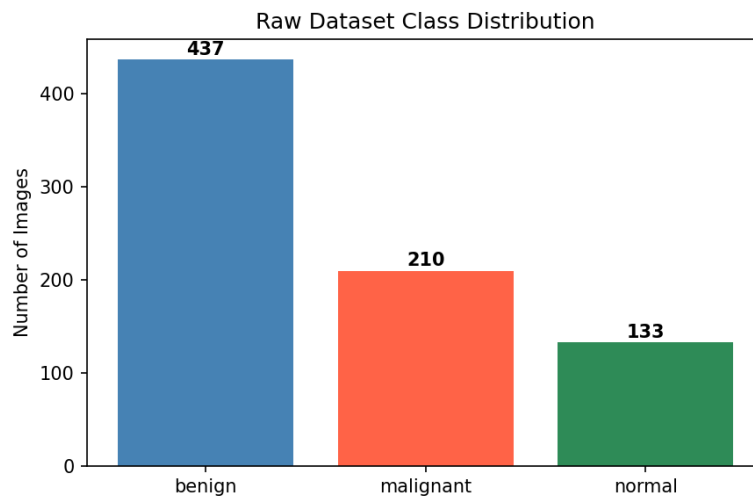


Figure 1. Class distribution of the raw dataset. Of the 780 images, 437 are benign, 210 malignant and 133 normal.

Images of the normal class contain no lesion, so their masks are empty and no bounding box can be derived from them. Consequently, although the detection problem is formally defined over three classes, in practice it corresponds to a two-class detection task, with the normal images acting as negative (background) examples that discourage false positives. This distinction matters for the interpretation of the results and is revisited in Section 6.

3.2. Formulation of the detection task

The objective is to learn a function f_{θ} that maps an image I to a finite set of predictions, $f_{\theta}(I) = \{(b_i, c_i, s_i)\}$, $i = 1, \dots, N$, where b_i denotes normalised box coordinates, $c_i \in \{\text{benign}, \text{malignant}\}$ the predicted class and $s_i \in [0, 1]$ the associated confidence score. The overlap between a predicted box and a ground-truth box is measured by the intersection over union (IoU):

$$\text{IoU}(b, \hat{b}) = \frac{|b \cap \hat{b}|}{|b \cup \hat{b}|} \quad (1)$$

3.3. Evaluation measures

For a given IoU threshold τ , predictions that exceed τ and carry the correct class are counted as true positives (TP), the remaining predictions as false positives (FP), and unmatched ground-truth boxes as false negatives (FN). Precision is defined as $P = TP / (TP + FP)$ and recall as $R = TP / (TP + FN)$. The area under the precision–recall

curve obtained by sweeping the confidence threshold gives the average precision (AP) of the corresponding class, where $\tilde{P}$ denotes the monotonically interpolated precision and R_n the n -th recall level:

$$AP = \sum_n (R_n - R_{n-1}) \cdot \tilde{P}(R_n) \quad (2)$$

The mean over all classes is denoted mAP. Following the PASCAL VOC convention, mAP@50 is computed at $\tau = 0.50$ [11]; following the COCO convention, mAP@50–95 is averaged over $\tau \in \{0.50, 0.55, \dots, 0.95\}$ [10]. Both are reported here. The second measure is considerably more sensitive to the precision of box localisation and therefore always takes lower values.

$$mAP@50 = \frac{1}{|C|} \sum_{c \in C} AP_c(\tau = 0.50) \quad (3)$$

4. Method

4.1. From masks to bounding boxes

The grayscale mask supplied with each image was binarised at an intensity threshold of 10 and its external contours were extracted. For each contour the minimum axis-aligned enclosing rectangle was computed, and components with an area below 20 pixels were discarded as noise. The resulting boxes were normalised into YOLO format as centre coordinates and dimensions. Empty label files were written for fully empty masks (the normal class), so that these images act as background examples during training. Figure 2 shows the labels produced in this way on samples drawn from the training subset.

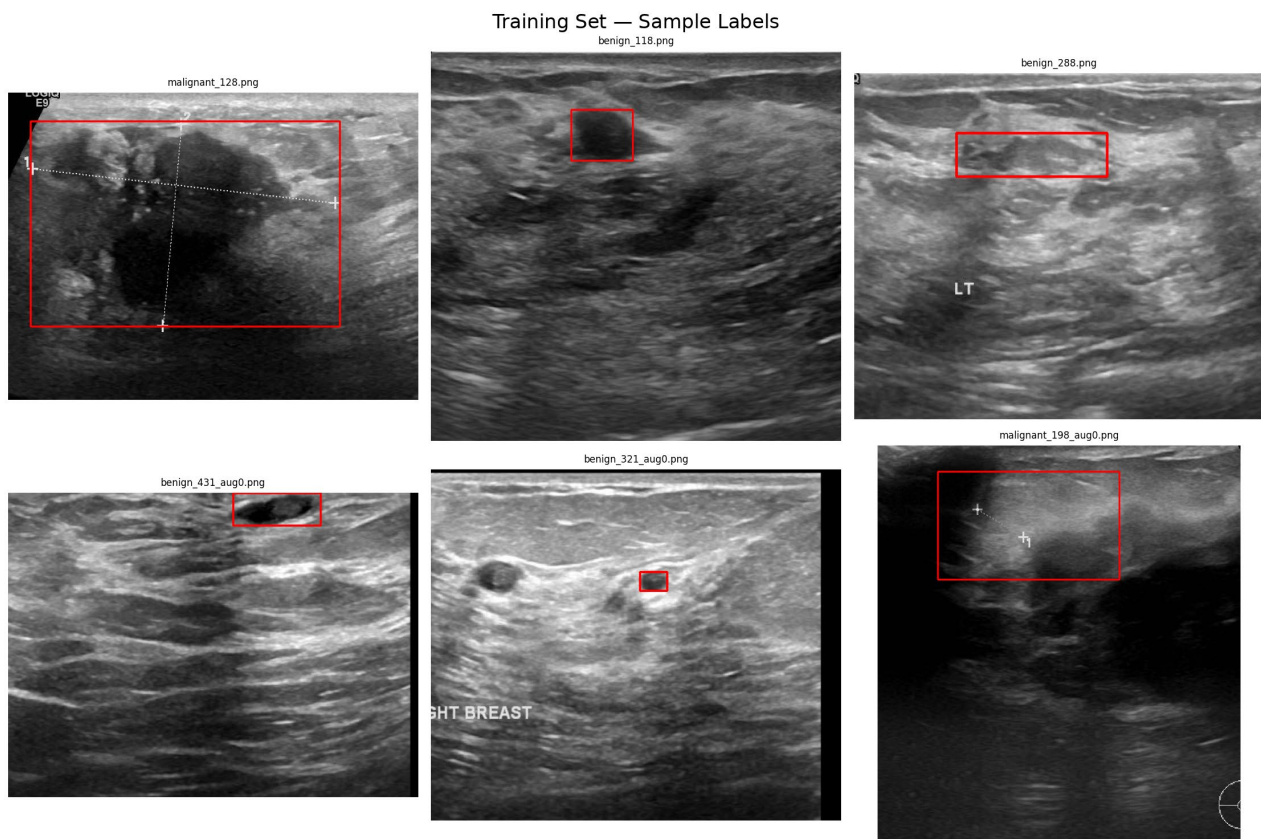


Figure 2. Sample images from the training subset with the bounding-box labels derived from their masks. The images carrying an '_aug0' suffix in the lower row are copies produced by offline augmentation.

This conversion has two intrinsic limitations. First, an axis-aligned rectangle also encloses surrounding normal tissue when the lesion boundary is irregular, which reduces the overlap between box and lesion, especially for malignant lesions with spiculated margins. Second, if a mask contains several disconnected components, several boxes are produced. In practice the number of boxes equalled the number of images (Table 1), which indicates that the lesions in this dataset were annotated as single connected components.

4.2. Partitioning and class balancing

Images were split independently within each class, using a fixed random seed (42), into 70% training, 15% validation and 15% test. The split was performed **before** augmentation, and augmentation was applied to the training subset only. This ordering structurally prevents the form of leakage in which the original of an image appears in training while an augmented copy of it appears in validation or test.

The augmentation pipeline comprised horizontal flipping ($p = 0.5$), random brightness and contrast adjustment ($p = 0.5$), rotation limited to $\pm 10^\circ$ ($p = 0.5$) and an affine transform ($\pm 5\%$ translation, 0.95–1.05 scaling, $p = 0.5$) [12]. Transformations were applied jointly to the image and its mask, and bounding boxes were recomputed from the transformed mask, so that label–image consistency was preserved. To reduce class imbalance, the number of augmented copies was made class-dependent: one additional copy for benign images, three for malignant images, and none for normal images. As a result the numbers of benign and malignant training instances became approximately equal (Table 1).

Table 1. Numbers of images and bounding-box instances in each subset after partitioning. Augmentation was applied to the training subset only. Normal images contain no lesion, so no boxes were generated for them; they were used as negative examples.

Subset	Benign	Malignant	Normal	Total images	Total boxes
Training (before augmentation)	307	148	95	550	455
Training (after augmentation)	614	592	95	1301	1206
Validation	65	31	19	115	96
Test	65	31	19	115	96

4.3. Model and training configuration

YOLOv8n, the smallest member of the YOLOv8 family, was selected as the detector. In its fused form the model has 73 layers, 3,006,233 trainable parameters and a cost of 8.1 GFLOPs at an input size of 640×640 . The choice is motivated by the size of the training set: larger models exhibit a pronounced tendency to overfit at this scale. Weights were initialised from the public yolov8n.pt checkpoint released by Ultralytics [5], which is pretrained for detection on COCO [10]; the whole network was then fine-tuned. Of the 355 parameter tensors of the model, 319 were transferred from this checkpoint, the detection head being re-initialised for the three-class configuration. The default online augmentations of the Ultralytics implementation (mosaic, horizontal flip, HSV jitter, scaling, translation and random erasing) were left enabled, with mosaic augmentation disabled for the final ten epochs.

The optimizer was left at the Ultralytics auto setting. For the dataset and batch size used here this selected AdamW [16] with an initial learning rate of 0.001429 and a first-moment decay of 0.9, and the trainer explicitly reported the nominal values $\text{lr}_0 = 0.01$ and $\text{momentum} = 0.937$ as ignored. Weight decay of 0.0005 was applied only to the convolutional and linear weight group; normalisation weights and biases were left undecayed. The settings listed in Table 2 are therefore those of the completed run rather than the requested defaults.

Table 2. Training configuration and hyperparameters. All values not listed are the defaults of Ultralytics version 8.4.71. The optimizer, initial learning rate and momentum were selected automatically by Ultralytics (optimizer='auto'); the nominal defaults $\text{lr}_0 = 0.01$ and momentum = 0.937 were consequently overridden. The values listed are the effective settings of the completed run.

Setting	Value	Setting	Value
Model	YOLOv8n	Optimizer	AdamW (auto-selected)
Number of parameters	3,006,233	Effective initial learning rate	0.001429
Input resolution	640 × 640	Momentum (β_1)	0.9
Batch size	16	Weight decay	0.0005
Epochs	50	Warm-up epochs	3
Early-stopping patience	15 (not triggered)	Loss weights (box/cls/DFL)	7.5 / 0.5 / 1.5
Random seed	0	Hardware	NVIDIA Tesla T4

5. Numerical Experiments

5.1. Experimental setting

All experiments were run in the Google Colab environment on a single NVIDIA Tesla T4 GPU. The software stack consisted of Python 3.12.13, PyTorch 2.11.0 (CUDA 12.8) and Ultralytics 8.4.71. Training for fifty epochs took approximately 0.40 hours (24 minutes). Model selection followed the default Ultralytics procedure. After every epoch the network is evaluated on the validation subset and a scalar fitness value is computed from the resulting metrics; the checkpoint attaining the highest fitness is retained as best.pt. In the version used here the fitness weights over [precision, recall, mAP@50 , mAP@50-95] are [0, 0, 0, 1], so fitness reduces to the validation mAP@50-95 . All results reported in Table 3 were obtained with this best.pt checkpoint, which corresponds to epoch 36. It is not the epoch at which the validation mAP@50 peaked (0.744 at epoch 37, Figure 4a): the two criteria disagree on this run, and the checkpoint selected on mAP@50-95 is the one used throughout. The test subset was evaluated once, in the final assessment only.

5.2. Training behaviour

Figure 3 shows the evolution of the box and classification losses. Both training losses decrease steadily, whereas the validation box loss plateaus in the range 1.10–1.20 after roughly epoch 25. The widening gap between the training and validation curves indicates a tendency to overfit, which is expected given the limited diversity of the training set. The sharp spike in the validation classification loss at the second epoch is associated with the high learning rate of the warm-up phase and disappears rapidly thereafter.

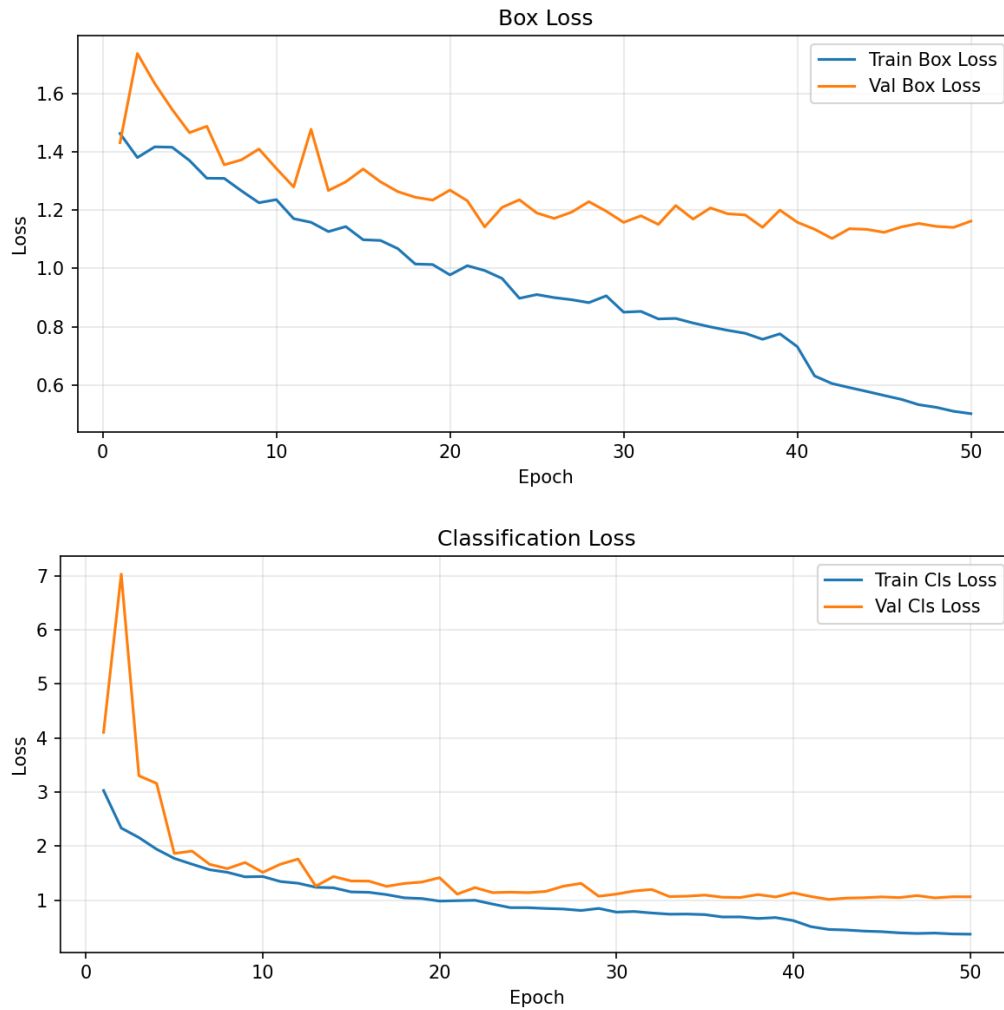


Figure 3. Training and validation losses against epoch: (a) box loss, (b) classification loss.

Figure 4 reports $\text{mAP}@50$ and $\text{mAP}@50-95$ measured on the validation subset. Both measures rise quickly during the first 15 epochs and then flatten with pronounced oscillation. The highest validation $\text{mAP}@50$, 0.744, was observed at epoch 37, although, as explained in Section 5.1, the checkpoint retained for evaluation was the one maximising $\text{mAP}@50-95$ rather than $\text{mAP}@50$. The amplitude of the oscillation is explained by the size of the validation subset, which contains only 96 instances: at this scale whether a handful of instances are matched or missed changes the measure visibly. The precision and recall curves in Figure 5 reflect the same behaviour, with the two measures fluctuating against each other throughout training. The early-stopping patience of 15 epochs was never triggered.

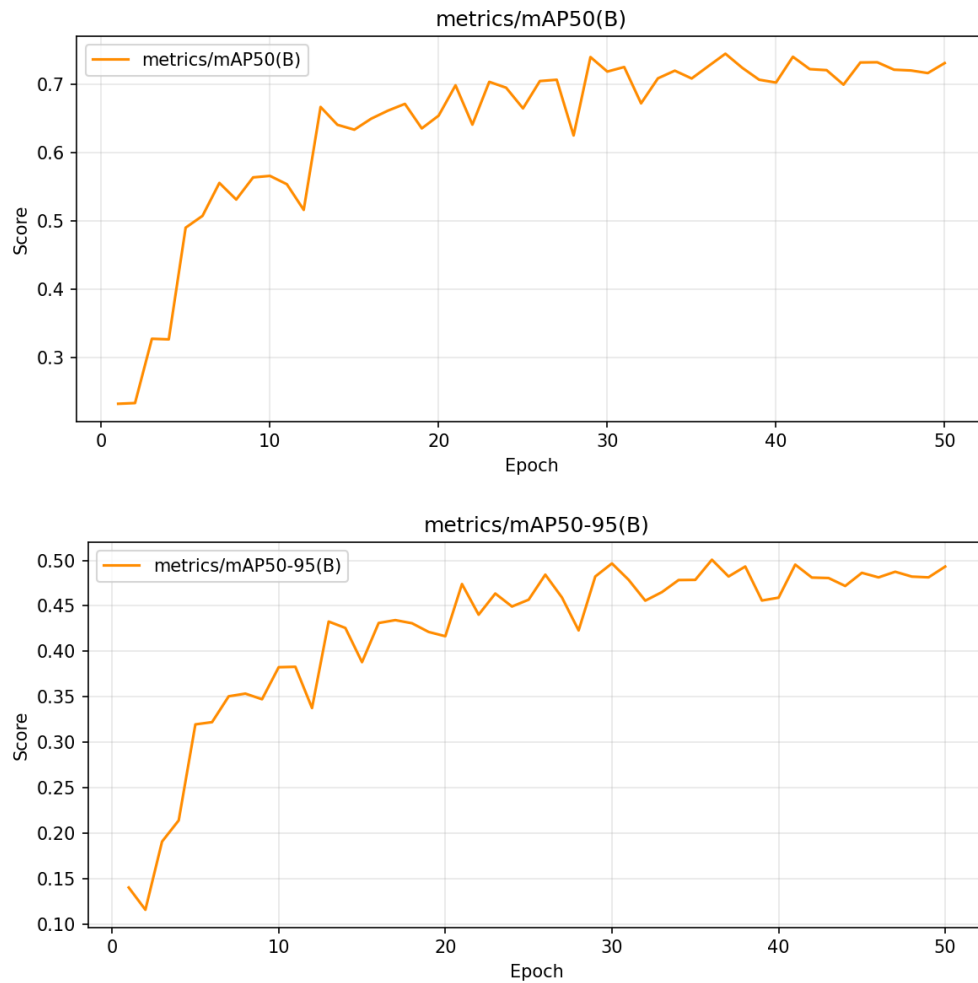


Figure 4. Average precision measured on the validation subset against epoch: (a) mAP@50, (b) mAP@50–95.

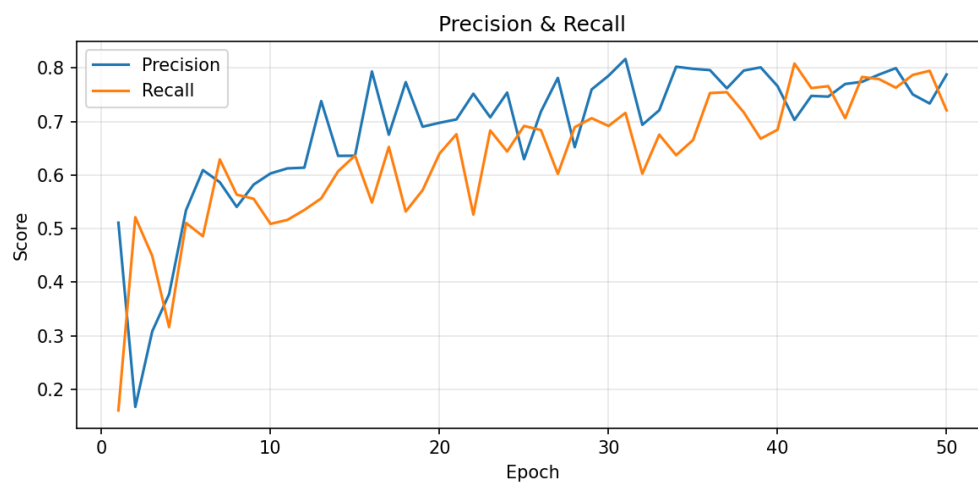


Figure 5. Precision and recall on the validation subset against epoch.

5.3. Validation and test results

Table 3 summarises the class-wise performance of the selected model on the validation and test subsets. On the test subset the overall mAP@50 was 0.737 and mAP@50–95 was 0.506. The differences between the two subsets

($\text{mAP@50} = 0.726$ on validation) should not be regarded as meaningful at this scale: both subsets contain 96 ground-truth boxes, and differences of this magnitude from a single split fall within the range of sampling variability.

Table 3. Performance of the selected model on the validation and test subsets. The normal class contains no bounding boxes and is therefore excluded from the average precision computation.

Subset	Class	Precision	Recall	mAP@50	mAP@50–95
Validation	Benign	0.928	0.799	0.839	0.634
Validation	Malignant	0.665	0.705	0.614	0.356
Validation	All	0.797	0.752	0.726	0.495
Test	Benign	0.761	0.783	0.817	0.585
Test	Malignant	0.679	0.645	0.656	0.428
Test	All	0.720	0.714	0.737	0.506

The precision and recall values in Table 3 are not read at a fixed operating point. Detections were produced with the Ultralytics validation defaults, which retain all predictions above a confidence floor of 0.001 after class-wise non-maximum suppression at an IoU threshold of 0.7. Precision and recall curves were then constructed over 1000 confidence levels, and the values reported are those at the confidence level maximising the smoothed mean F1 score across classes. The mAP values, in contrast, are integrals over the full precision–recall curve and are therefore independent of any single confidence threshold. A deployment would require this operating point to be fixed deliberately — and, for a screening-support application, biased towards recall — rather than inherited from an F1-optimal criterion that weights false positives and false negatives equally.

The most notable finding is the gap between the classes: on the test subset mAP@50 reaches 0.817 for benign lesions but only 0.656 for malignant ones, and the gap widens under the stricter mAP@50–95 measure (0.585 versus 0.428). Three explanations are plausible. First, the raw dataset contains roughly half as many malignant as benign examples; offline augmentation equalises the counts but adds no new information, merely replicating transformations of the examples already present. Second, malignant lesions typically have irregular, spiculated boundaries that an axis-aligned box represents poorly, which lowers the probability of a match at high IoU thresholds. Third, malignant lesions frequently produce pronounced acoustic shadowing, and whether the shadowed region is included in the lesion can introduce a systematic offset between the ground truth and the prediction.

In terms of throughput, the model reached an average inference time of 5.8 ms per image on the test subset (6.8 ms preprocessing, 3.8 ms postprocessing). These figures indicate a computational cost compatible with real-time operation on a single T4 GPU, which places the model at a scale suitable for an application running concurrently with an ultrasound examination.

5.4. Qualitative assessment and error analysis

To examine the behaviour of the model beyond the aggregate measures, inference was performed on a malignant image at a confidence threshold of 0.25, the Ultralytics prediction default. The subset membership of this image was not recorded at the time the figure was produced. We subsequently identified the file and regenerated the partition from the same fixed random seed, which places it in the test subset; six images appearing in the training-set illustration of Figure 2 were used as anchors and all fell in the training partition of the regenerated split, which is consistent with, though not proof of, an exact reconstruction of the original partition. In the case shown in Figure 6 the model produced no detection: the lesion was missed entirely. The lesion boundary separates only weakly from the surrounding tissue and marked acoustic shadowing is present. The quantitative evidence for this failure mode comes from the test subset in Table 3, where a recall of 0.645 over 31 malignant instances corresponds to approximately eleven malignant lesions missed at the F1-optimal operating point. False negatives of this kind are clinically costlier than false positives, since missing a malignant lesion has far more serious consequences than an unnecessary request for further examination. Lowering the confidence threshold would reduce their number at the expense of precision, but this trade-off must be fixed on a large and independent test set rather than inferred from

individual cases. The threshold of 0.25 used here is the prediction default and is deliberately distinct from the threshold-free protocol underlying Table 3.

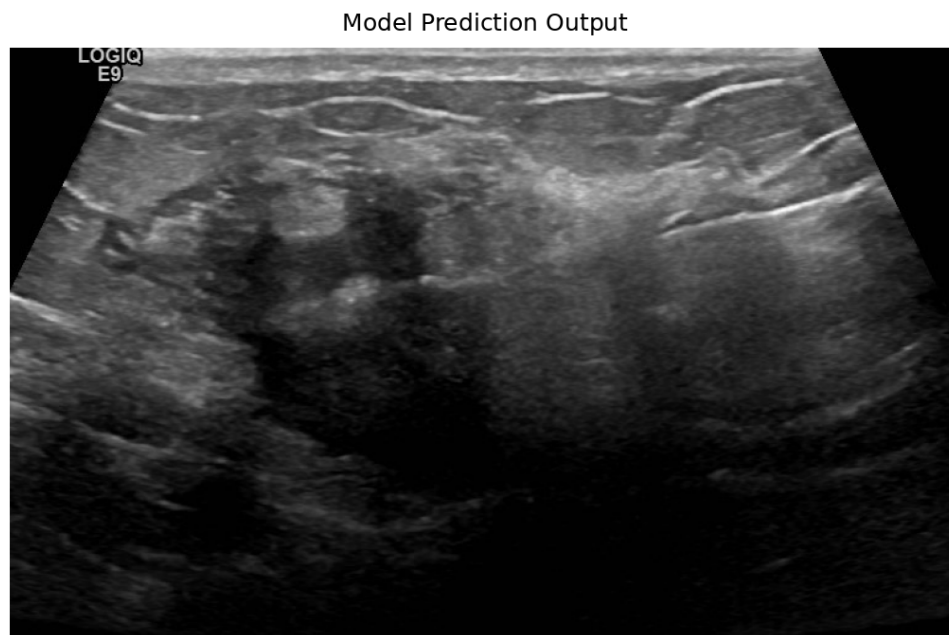


Figure 6. Inference on a malignant image assigned to the test subset by regenerating the partition from the recorded random seed. No bounding box was produced at a confidence threshold of 0.25; the case illustrates the false-negative behaviour of the model.

6. Discussion and Conclusion

This study trained a YOLOv8n-based model for lesion detection in breast ultrasound images and documented every step of data preparation, training and evaluation. The values obtained on the held-out test subset, $\text{mAP}@50 = 0.737$ and $\text{mAP}@50-95 = 0.506$, show that a lightweight detector trained on a small dataset can attain a reasonable localisation performance. Performance on the clinically decisive malignant class ($\text{mAP}@50 = 0.656$, recall 0.645), however, is not adequate for decision-support use.

The limitations should be stated plainly. **(a)** The dataset is single-centre and comprises 780 images; whether the results generalise to other devices, protocols and populations was not tested. **(b)** The split was performed at image level rather than at patient level, so several images from the same patient may fall into different subsets. The reported presence of duplicate images in BUSI [7] increases this risk and means that the reported figures may be somewhat optimistic. **(c)** Results are reported for a single training run on a single split; although the split itself is reproducible from the recorded seed, no cross-validation and no repetition over different seeds were carried out, so confidence intervals cannot be given. **(d)** No comparative baseline (for instance a two-stage detector or other YOLO versions) was evaluated. **(e)** Since the normal class carries no bounding boxes, the model in fact learns a two-class detection task, and the contribution of normal images as background examples was not measured separately. **(f)** This is a computational feasibility study; no clinical validation was performed at any stage and the model presented here must not be used for diagnostic purposes.

Three directions stand out for future work. First, patient-level stratified cross-validation combined with repetition over random seeds would attach confidence intervals to the reported figures. Second, class imbalance can be addressed not only through augmentation but also through loss-level reweighting such as focal loss [15], and the contribution of the two mechanisms can be separated. Third, given that axis-aligned boxes represent irregular lesion shapes poorly, instance-segmentation heads that learn detection and segmentation jointly may produce more informative output, particularly for the malignant class. Finally, cross-dataset evaluation over several public datasets would be decisive for measuring generalisation.

CRedit authorship contribution statement. D.ÖĞÜZ: Conceptualization, Methodology, Software, Data curation, Visualization, Writing – original draft. S.GENÇ: Supervision, Methodology, Validation, Writing – review & editing. Ö.Ş.SÖNMEZ : data interpretation.

Funding. This work was supported by TÜBİTAK under the 2209-A University Students Research Projects Support Programme (project no. 1919B012467372).

Declaration of competing interest. The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Ethics statement. The study uses only a publicly available, de-identified image dataset [6] that was published with ethical approval and patient consent obtained at source. No additional ethics committee approval was therefore required. The study is not a diagnostic validation study and the model presented is not intended for clinical use.

Data and code availability. The BUSI dataset used in this study is publicly available and can be obtained through reference [6]; it is not redistributed here. The notebook containing the data preparation, training and evaluation steps, the trained model weights, the epoch-level training records and the figure sources are deposited at <https://doi.org/10.57967/hf/10143>.

Declaration of generative AI use. A large language model assistant was used during the preparation of this manuscript for language editing, translation, literature reviews and code assistance. Generative AI tools are not listed as authors; the authors take full responsibility for every claim, analysis and citation in the paper.

References

- [1] F. Bray, M. Laversanne, H. Sung, J. Ferlay, R. L. Siegel, I. Soerjomataram, and A. Jemal, “Global cancer statistics 2022: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries,” *CA Cancer J. Clin.*, vol. 74, no. 3, pp. 229–263, 2024, doi: 10.3322/caac.21834.
- [2] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. W. M. van der Laak, B. van Ginneken, and C. I. Sánchez, “A survey on deep learning in medical image analysis,” *Med. Image Anal.*, vol. 42, pp. 60–88, 2017, doi: 10.1016/j.media.2017.07.005.
- [3] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, 2016, pp. 779–788, doi: 10.1109/CVPR.2016.91.
- [4] J. Terven, D.-M. Córdova-Esparza, and J.-A. Romero-González, “A comprehensive review of YOLO architectures in computer vision: From YOLOv1 to YOLOv8 and YOLO-NAS,” *Mach. Learn. Knowl. Extr.*, vol. 5, no. 4, pp. 1680–1716, 2023, doi: 10.3390/make5040083.
- [5] G. Jocher, A. Chaurasia, and J. Qiu, “Ultralytics YOLOv8,” software version 8.4.71, 2023. [Online]. Available: <https://github.com/ultralytics/ultralytics>
- [6] W. Al-Dhabyani, M. Gomaa, H. Khaled, and A. Fahmy, “Dataset of breast ultrasound images,” *Data Brief*, vol. 28, art. no. 104863, 2020, doi: 10.1016/j.dib.2019.104863.
- [7] A. Pawłowska, P. Karwat, and N. Żolek, “Letter to the editor. Re: ‘Dataset of breast ultrasound images by W. Al-Dhabyani, M. Gomaa, H. Khaled & A. Fahmy, Data in Brief, 2020, 28, 104863,’” *Data Brief*, vol. 48, art. no. 109247, 2023, doi: 10.1016/j.dib.2023.109247.
- [8] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, 2017, doi: 10.1109/TPAMI.2016.2577031.
- [9] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional networks for biomedical image segmentation,” in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015, pp. 234–241, doi: 10.1007/978-3-319-24574-4_28.
- [10] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, “Microsoft COCO: Common objects in context,” in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 740–755, doi: 10.1007/978-3-319-10602-1_48.
- [11] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, “The PASCAL Visual Object Classes (VOC) challenge,” *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, 2010, doi: 10.1007/s11263-009-0275-7.
- [12] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, and A. A. Kalinin, “Albumentations: Fast and flexible image augmentations,” *Information*, vol. 11, no. 2, art. no. 125, 2020, doi: 10.3390/info11020125.
- [13] C. Shorten and T. M. Khoshgoftaar, “A survey on image data augmentation for deep learning,” *J. Big Data*, vol. 6, art. no. 60, 2019, doi: 10.1186/s40537-019-0197-0.
- [14] X. Li, W. Wang, L. Wu, S. Chen, X. Hu, J. Li, J. Tang, and J. Yang, “Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection,” in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 33, 2020, pp. 21002–21012.
- [15] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, 2017, pp. 2980–2988, doi: 10.1109/ICCV.2017.324.
- [16] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” in *Proc. 7th Int. Conf. Learn. Represent. (ICLR)*, New Orleans, LA, USA, 2019. doi: 10.48550/arXiv.1711.05101.