Scientific Research Publishing

Journal of Computer and Communications, 2026, 14(11), 287-301
https://www.scirp.org/journal/jcc
ISSN Online: 2327-5227
ISSN Print: 2327-5219

APU-Net: A Task-Adapted U-Net for Fast Phase-to-Amplitude Acoustic Field Prediction

Molin Fang[1], Xuewei Wang[1*], Changbo Xu[1], Wanggong Yang[2], Changxu Wang[3]

[1]College of Information Engineering, Beijing Institute of Graphic Communication, Beijing, China
[2]College of New Media, Beijing Institute of Graphic Communication, Beijing, China
[3]The Sino-British College, University of Shanghai for Science and Technology, Shanghai, China
Email: milyofficial155@gmail.com, *wangxuewei@bigc.edu.cn, xuchangbo@bigc.edu.cn, ywg@bigc.edu.cn, abc731864539@gmail.com

How to cite this paper: Fang, M.L., Wang, X.W., Xu, C.B., Yang, W.G. and Wang, C.X. (2026) APU-Net: A Task-Adapted U-Net for Fast Phase-to-Amplitude Acoustic Field Prediction. *Journal of Computer and Communications*, **14**, 287-301.
https://doi.org/10.4236/jcc.2026.1411013

Received: November 2, 2026
Accepted: November 22, 2026
Published: November 25, 2026

Copyright © 2026 by author(s) and Scientific Research Publishing Inc.
This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).
http://creativecommons.org/licenses/by/4.0/

Open Access

Abstract

Direct prediction of a target-plane amplitude distribution from a source-plane phase distribution is an important task in acoustic field modeling. For a fixed acoustic configuration, this prediction must account for spatial variations across the target field while allowing repeated evaluations at low computational cost. We propose Acoustic Prediction U-Net (APU-Net) to predict the target-plane amplitude distribution from the source-plane phase distribution. The network adopts an encoder-decoder structure with normalized convolutional blocks, bilinear upsampling in the decoder, explicit feature-map alignment, and a 512-channel bottleneck to control model capacity. Tests on the MNIST-derived dataset yield an MSE of 0.000451 and a PSNR of 33.46 dB. Compared with the standard U-Net, APU-Net reduces MSE by 79.5% and improves PSNR by 6.89 dB, with a comparable average computation time. It also runs more than 20 times faster than the recorded ASM implementation at all tested resolutions under the same runtime environment. These results show that APU-Net can predict target-plane amplitude distributions from source-plane phase inputs with improved accuracy and substantially lower computational cost than the recorded ASM implementation.

Keywords

Acoustic Hologram, Deep Learning, Acoustic Field Reconstruction, U-Net, Wave Propagation

1. Introduction

Acoustic manipulation has found applications in bioengineering [1]-[3], lab-on-a-chip systems [4], material delivery [5], and biomedical research [6]-[10]. In

acoustic holography, the pressure field can be shaped for purposes such as microengineering [11] [12] and energy harvesting [13] [14]. In applications involving dynamic field control, many candidate phase patterns may need to be evaluated under the same acoustic configuration, making computational efficiency an important consideration.

Phase modulation [15] can be implemented using passive transmission holograms or actively controlled phased array transducers (PATs). Passive holograms are commonly fabricated using techniques such as 3D printing [16], whereas PATs provide electronic control over the emitted phase [17].

In a passive acoustic hologram, the desired phase delay is encoded in the structure during fabrication. Melde *et al.* demonstrated that a single transducer combined with a transmission hologram can produce complex target fields [16]. This arrangement requires only a simple drive configuration and can provide high spatial resolution over a large aperture. However, the resulting field pattern depends on the accuracy of fabrication, and the structure is designed for a predefined field rather than subsequent dynamic updates.

Unlike passive holograms, PATs allow the emission phase of individual array elements to be adjusted [17]. This enables the target field to be changed dynamically. The spatial degrees of freedom available for field control depend on factors such as the number of array elements, their spacing, and the aperture size. When studying dynamic PAT operation, computational methods may therefore need to evaluate multiple candidate phase patterns while keeping the acoustic configuration unchanged.

For forward acoustic field calculation, the Angular Spectrum Method (ASM) propagates a specified source-plane phase distribution to obtain the corresponding target-plane acoustic response [18]. Its numerical accuracy can be affected by the propagation implementation. Zeng and McGough investigated window-truncation errors in spatial-domain propagation and aliasing effects in spectral-domain propagation for near-field pressure simulation [19]. ASM and Green-function formulations are both established approaches for calculating forward acoustic fields.

Deep neural networks have also been applied to mappings between different acoustic field representations [20]-[23]. For example, Zhong *et al.* proposed AcousNet for dynamic three-dimensional field generation using phased arrays [23], while Lin *et al.* developed a U-Net-based approach for phase prediction [21]. The problem addressed in this work differs in the direction of prediction. It maps a source-plane phase distribution to a target-plane amplitude distribution. The input contains spatially distributed phase information, while the output is a continuous amplitude map. Reconstructing this map requires the network to account for both its overall spatial structure and local details. These requirements motivate the use of an encoder-decoder architecture, with particular attention to normalization, upsampling, feature alignment, and bottleneck capacity.

APU-Net addresses this model-design problem through a task-adapted U-Net

architecture for direct P_S -to- A_E prediction. Our study makes three model-centered contributions. First, APU-Net integrates normalized convolutional blocks, interpolation-based upsampling, explicit decoder-map alignment, and a 512-channel bottleneck into a unified phase-to-amplitude predictor [24] [25]. Second, the complete architecture achieves a substantially better accuracy-efficiency balance than the evaluated CNN, U-Net, and IU-Net models under the recorded training protocol. Third, its operating behavior is characterized through multi-resolution timing, controlled phase perturbations, and evaluation on two additional image-pattern distributions.

Section 2 defines the phase-to-amplitude prediction task and data flow. Section 3 presents the APU-Net architecture, its design rationale, and the evaluation metrics. Section 4 evaluates prediction accuracy, optimization behavior, multi-resolution efficiency, response to phase perturbations, cross-pattern transfer, and comparisons with neural models. Section 5 concludes the paper.

2. Phase-to-Amplitude Prediction

Figure 1 shows the phase-to-amplitude task for a PAT. Let A_E denote the expected target-plane amplitude distribution, P_S the source-plane phase distribution, and A_C the amplitude distribution predicted by APU-Net. The network learns F: $P_S \rightarrow A_E$ and minimizes the discrepancy between A_C and A_E. This mapping is treated as a cross-representation dense-regression problem in which the encoder must aggregate the distributed structure of P_S and the decoder must recover the spatial detail of A_C. The experiments use one fixed two-stage prediction workflow.

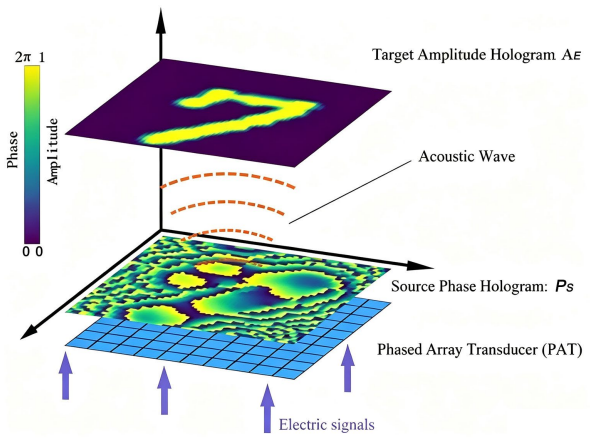


Figure 1. Phase-to-amplitude prediction for a phased array transducer.

Physics-based methods such as angular spectrum method (ASM) and Green-function propagation compute acoustic fields from specified source conditions [18] [19]. APU-Net learns the P_S -to- A_E mapping represented by the ASM-propagated paired dataset and executes it through a single encoder-decoder forward pass. The model is evaluated by its agreement with the target amplitude dis-

tribution A_E and by its per-image computation time, while ASM serves as the physics-based runtime comparator in Section 4.

The fixed two-stage prediction workflow starts with the previously developed deep learning-based acoustic reconstruction model, which we call the phase-generation model. Stage 1 applies this model to the prescribed target pattern A_{E_ORI} to produce the source-plane phase distribution P_S. Propagating P_S with ASM under the same acoustic configuration yields the target-plane amplitude distribution A_E, which approximates A_{E_ORI} and serves as the supervised target. Stage 2 applies APU-Net to P_S to produce the predicted amplitude distribution A_C. Because each training pair (P_S, A_E) is obtained directly through ASM propagation, APU-Net is trained on physically propagated field data rather than on arbitrary image pairs. The purpose of APU-Net is to replace the computationally expensive ASM evaluation with a fast learned approximation under the fixed acoustic configuration.

The acoustic configuration considered in this study is a 50 × 50 element phased transducer array with an aperture of 50 mm × 50 mm, operating at a frequency of 1 MHz in water (sound speed 1500 m/s). The imaging plane is located at a distance of 10 cm from the source plane.

3. APU-Net Method

3.1. Model Architecture

APU-Net follows the multiscale encoder-decoder structure of U-Net [26]. For the P_S-to-A_E prediction task, the encoder extracts spatial features at different scales from the source-plane phase distribution. These features are then passed to the decoder, where skip connections help preserve local information during the reconstruction of the target-plane amplitude distribution. Several modifications are introduced to the original U-Net, including changes to the normalization layers, decoder upsampling, feature-map alignment, and bottleneck width.

The overall architecture is illustrated in **Figure 2**. BatchNorm2d is applied after each convolution to normalize the intermediate feature maps in both the encoder and decoder [24]. In the decoder, bilinear interpolation is used to recover spatial resolution instead of transposed convolution. Padding is also applied when necessary to match the spatial dimensions of decoder features with those of the corresponding encoder features before concatenation [25]. The bottleneck is set to 512 channels to provide sufficient feature representation without excessively increasing model capacity. The effects of these design choices are examined in Section 4.

These modifications were made with the characteristics of phase-to-amplitude prediction in mind. The network performs regression on a continuous amplitude distribution within [0, 1], making the consistency of intermediate feature representations an important consideration. BatchNorm2d is therefore used throughout the network. The phase input also contains interference fringes with relatively stable statistical properties, which makes channel-wise normalization suitable for

this task. For decoder upsampling, bilinear interpolation provides a direct way to increase spatial resolution without introducing additional learned upsampling kernels. This choice also avoids potential checkerboard patterns associated with transposed convolution [25]. The bottleneck width was set to 512 channels to retain sufficient information during feature extraction. This is relevant to ASM propagation, where the target field depends on contributions from the entire source plane rather than only its local neighborhood. At the same time, an excessively large bottleneck could increase the risk of overfitting on the available training data. Another consideration is spatial alignment: the 50 × 50 input size does not divide evenly through successive pooling operations. Padding is therefore needed before skip connections are applied, so that features from different network levels remain spatially aligned.

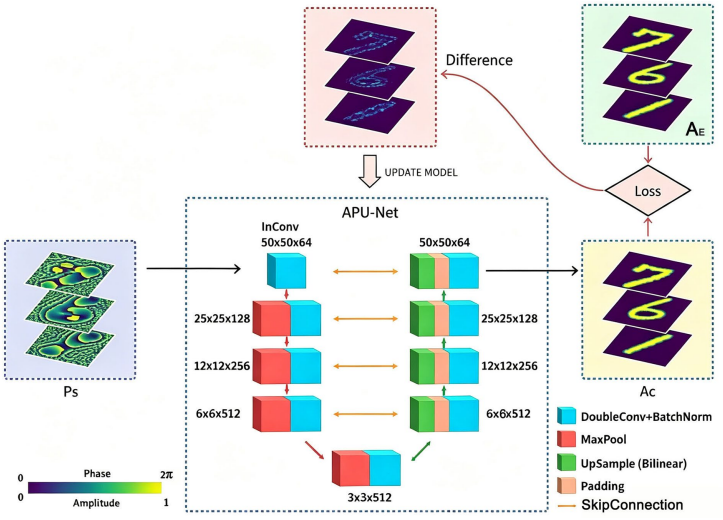


Figure 2. APU-Net architecture. Given the source-plane phase distribution P_S as input, the network produces the corresponding target-plane amplitude distribution A_C.

3.2. Evaluation Metrics

To assess prediction accuracy and computational efficiency, we use Mean Squared Error (MSE), Dice Coefficient, Global Correct rate, Structural Similarity Index (SSIM), Peak Signal-to-Noise Ratio (PSNR), and average computation time.

MSE summarizes the squared pixel differences between the predicted amplitude distribution A_C and the expected distribution A_E over N pixels:

$$\mathrm{MSE} = \frac{1}{N}\sum\left(A_C - A_E\right)^2 \tag{1}$$

The Dice Coefficient evaluates the overlap between the predicted region A and the expected region B:

$$\mathrm{Dice} = \frac{2\left|A \cap B\right|}{\left|A\right| + \left|B\right|} \tag{2}$$

The Global Correct rate (GC) measures the global consistency between the predicted amplitude distribution A_C and the expected distribution A_E:

$$\mathrm{GC} = \frac{N_{correct}}{N_{total}} \tag{3}$$

Structural Similarity Index (SSIM) evaluates the structural similarity between A_C and A_E through luminance, contrast, and structure terms:

$$\mathrm{SSIM}(x,y) = \frac{\left(2\mu_x\mu_y + C_1\right)\left(2\sigma_{xy} + C_2\right)}{\left(\mu_x^2 + \mu_y^2 + C_1\right)\left(\sigma_x^2 + \sigma_y^2 + C_2\right)} \tag{4}$$

Peak Signal-to-Noise Ratio (PSNR) measures the pixel-wise error under the normalized amplitude representation, where MAX is the maximum possible amplitude value:

$$\mathrm{PSNR} = 10\log_{10}\left(\frac{\mathrm{MAX}^2}{\mathrm{MSE}}\right) \tag{5}$$

Implementation details of these metrics are as follows. Both the Dice Coefficient and GC are computed after binarizing the predicted and expected fields with a threshold of 0.5. SSIM accounts for luminance, contrast, and structural information, and it is computed here with the default scikit-image implementation using a 7 × 7 window.

4. Results

APU-Net is implemented in PyTorch and trained on an NVIDIA GeForce RTX 4060 graphics processing unit (GPU). The training configuration consists of 150 epochs, an initial learning rate of 0.05, a batch size of 8, stochastic gradient descent, and MSE loss. Four aspects of the model are examined: prediction fidelity, millisecond-level inference, sensitivity to controlled phase perturbations, and transfer across image-pattern distributions.

The dataset is generated through a fixed pair-generation pipeline using 60,000 MNIST images. Each grayscale image is first normalized from [0, 255] to [0, 1] by linear scaling, giving the prescribed target-plane amplitude distribution A_{E_ORI}. The phase-generation model then uses A_{E_ORI} to produce a source-plane phase distribution P_S within [0, 2π]. For each P_S, forward propagation is performed with the ASM under the fixed acoustic configuration. The corresponding target-plane amplitude distribution A_E is obtained from this propagation, and the resulting (P_S, A_E) pairs form the supervised dataset used to train APU-Net.

Throughout data generation, the same ASM implementation and acoustic parameters are applied to every source-plane phase distribution. Network training and evaluation use the propagated amplitude distributions A_E as reference targets, rather than the originally prescribed patterns A_{E_ORI}. In this way, each reference target is obtained directly from the acoustic propagation of its corresponding source-plane phase distribution.

The dataset is organized into ordered pairs (P_S, A_E). For each pair, APU-Net uses P_S as the input and produces a predicted amplitude distribution A_C. The prediction is optimized by minimizing the MSE between A_C and A_E. The data

are split into training, validation, and test sets in proportions of 0.7, 0.15, and 0.15, respectively. The same pair-generation pipeline is also applied to FashionMNIST and Char74K for cross-dataset testing.

4.1. Prediction Accuracy

Figure 3 compares the predicted amplitude distribution A_C with the expected distribution A_E. The predictions preserve the principal digit shapes, boundaries, and amplitude patterns. APU-Net records a PSNR of 33.46 dB, an SSIM of 0.9901, a Dice Coefficient of 0.9779, and a Global Correct rate of 99.45% on the test set. Together, these metrics quantify the pixel-level and structural fidelity of the complete APU-Net architecture in the paired prediction task.

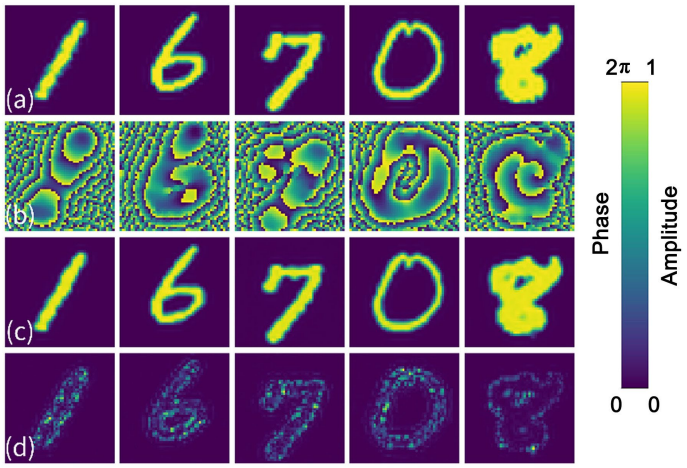


Figure 3. Phase-to-amplitude prediction results. (a) Expected target-plane amplitude distribution A_E; (b) source-plane phase distribution P_S; (c) predicted amplitude distribution A_C; and (d) difference between A_E and A_C.

Figure 4 presents the changes in training and validation losses during the 150-epoch training process. Both losses decline as training progresses, following broadly similar patterns. At the end of training, the loss values are 0.00045 for the training set and 0.00046 for the validation set. The small difference between these values suggests that the model maintains consistent performance on the two sets. Training is completed in 109 minutes.

4.2. Multi-Resolution Inference Efficiency

Table 1 compares the per-image computation times of APU-Net and ASM at different image resolutions. Resolution is defined by the pixel dimensions of the input image. To obtain the higher-resolution inputs, the base images were upsampled to the corresponding sizes using Lanczos interpolation. Both methods were tested on the workstation described above, using the same hardware and software environment. The input resolutions and timing units were also kept consistent across the two methods.

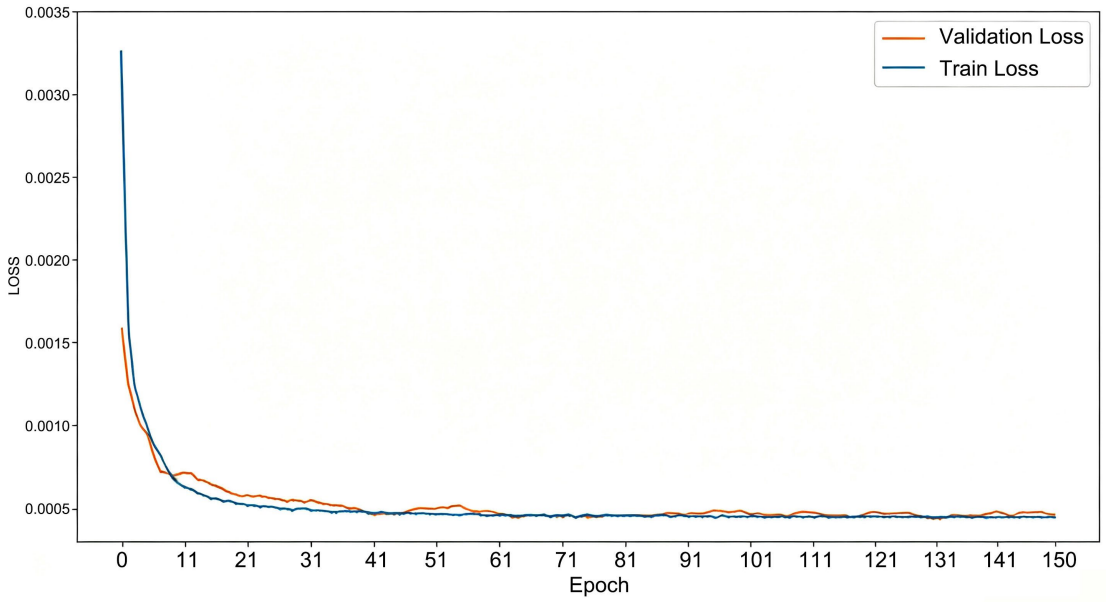


Figure 4. Training and validation loss.

At 50 × 50 resolution, APU-Net takes 1.93 ms per image, compared with 53.34 ms for ASM. When the resolution increases to 1024 × 1024, the corresponding times are 5.21 ms and 175.62 ms. The results show that ASM takes more than 20 times as long as APU-Net at all reported resolutions. As illustrated in **Figure 5**, the amplitude distributions predicted by APU-Net also show good visual agreement with those obtained through ASM propagation at different resolutions.

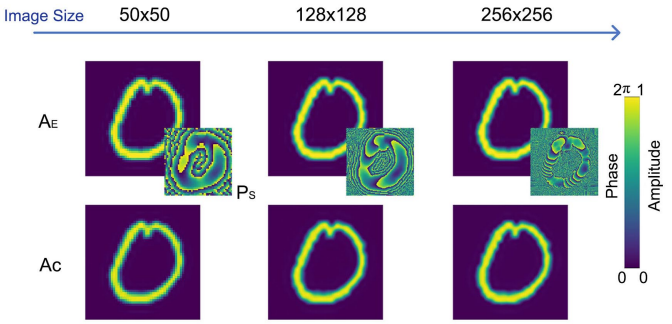


Figure 5. Comparison between the APU-Net-predicted amplitude A_C and the ASM-propagated amplitude A_E at different image resolutions (from 50 × 50 to 256 × 256).

Table 1. Computation time of APU-Net and ASM.

Resolution	APU-Net	ASM
50 × 50	1.93 ms	53.34 ms
128 × 128	1.96 ms	62.72 ms
256 × 256	2.13 ms	81.21 ms
512 × 512	3.17 ms	134.17 ms
1024 × 1024	5.21 ms	175.62 ms

4.3. Robustness

We test the complete APU-Net model using two types of perturbations to the source-plane phase input P_S : Gaussian noise and circular zero-phase replacement. The modified input is denoted by P_S' . These experiments are designed to examine how the model's predictions change when the phase input is altered.

In the first test, Gaussian noise is added to P_S . Specifically, zero-mean Gaussian noise with a standard deviation of 25 is applied in the 0 - 255 pixel-value domain, corresponding to approximately 0.62 rad, or 9.8% of the full 2π phase range. A random mask determines the pixels to be corrupted, with four corruption ratios considered: 20%, 40%, 60%, and 80%. Noise is applied only to the selected pixels, while the remaining pixels are left unchanged. The resulting values are clipped to [0, 255] without phase wrapping. As shown in Figure 6, the predicted amplitude distribution A_C preserves its main spatial features under the perturbations considered. However, the predicted amplitude values tend to decrease as the corruption ratio increases.

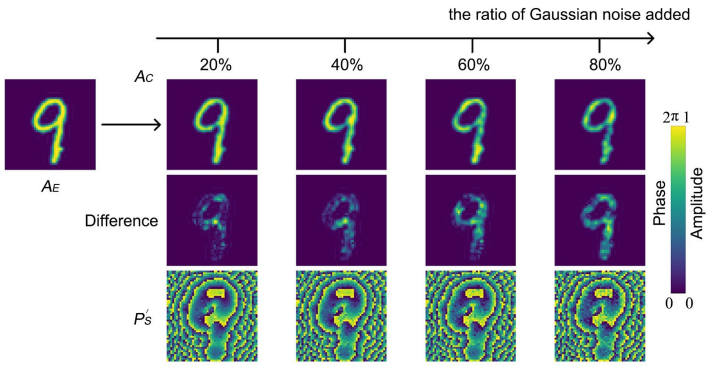


Figure 6. APU-Net predictions under different levels of Gaussian noise.

To evaluate the model's robustness to localized phase-data corruption, circular regions with randomly selected locations and diameters of 10 - 20 pixels are introduced into the input phase distribution. Within each region, the phase values are replaced with zero. As shown in Figure 7, the annotation "Dot Location" specifies the center coordinates and diameter of each circular region. For example, (41, 11) d = 10 indicates a circular region with a diameter of 10 pixels centered at coordinates (41, 11).

As shown in Figure 7, APU-Net retains the global pattern despite these localized perturbations. Dark traces remain near several modified regions, while the principal digit shapes remain visible. These results characterize the model's robustness to localized phase-data corruption and its ability to preserve the overall pattern when portions of the input phase data are corrupted.

4.4. Generalization

We train APU-Net on MNIST pairs and evaluate the same model on Char74K and FashionMNIST pairs generated by the same pipeline. Figure 8 shows an average

PSNR of 31.64 dB on Char74K and 23.78 dB on FashionMNIST. The results characterize how the learned APU-Net representation transfers from handwritten digits to two additional image-pattern distributions while the acoustic configuration and pair-generation procedure remain fixed.

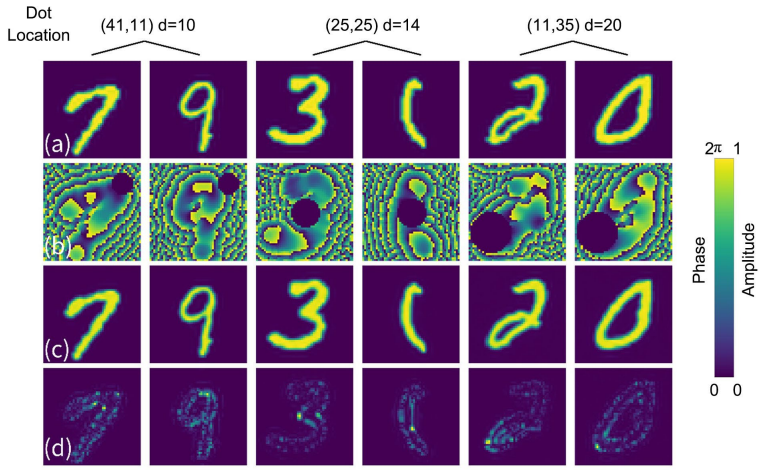


Figure 7. Prediction under localized zero-phase replacement. (a) Unperturbed reference amplitude distribution A_E; (b) perturbed phase distribution P_S'; (c) predicted amplitude distribution A_C; and (d) difference between A_E and A_C.

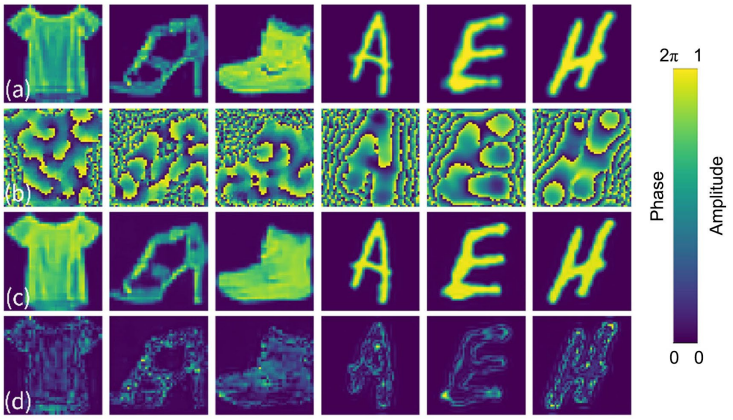


Figure 8. Cross-dataset prediction results. (a) Expected amplitude distribution A_E; (b) source-plane phase distribution P_S; (c) predicted amplitude distribution A_C; and (d) difference between A_E and A_C.

4.5. Accuracy-Efficiency Comparison with Models

Table 2 compares a convolutional neural network (CNN), U-Net, IU-Net [27], and APU-Net after 150 training epochs on the MNIST-based pairs. IU-Net is included as a representative domain-related U-Net variant because it was developed for rapid acoustic hologram reconstruction. Its closely related reconstruction setting, ASM-based phase-data construction, U-shaped encoder-decoder with skip connections, adaptive learning-rate scheduling, MNIST and Char74K evaluations,

and reported accuracy-efficiency metrics make it a relevant domain comparison model alongside the generic CNN and standard U-Net. IU-Net was re-implemented and retrained from scratch on the same MNIST-based pairs under the identical training protocol used for APU-Net. Under the common training protocol used here, APU-Net reduces MSE from 0.002204 to 0.000451 relative to U-Net, a reduction of 79.5%, and increases PSNR from 26.57 dB to 33.46 dB, while the average computation time remains comparable at 1.93 ms versus 1.91 ms. Compared with IU-Net, APU-Net achieves a 50.8% lower MSE and a 3.08 dB higher PSNR. Their computation times are nearly identical, at 1.93 ms for APU-Net and 1.97 ms for IU-Net. Among the neural models included in the comparison, APU-Net also obtains the highest Dice Coefficient, Global Correct rate, and SSIM. Taken together, these results show that APU-Net provides better prediction accuracy than both the standard U-Net and the tested IU-Net implementation, while maintaining a similar computation time. When retraining IU-Net, we retained its adaptive learning rate strategy and followed the same experimental protocol for the remaining hyperparameters. This allows IU-Net to use its original learning rate scheme while keeping the other training conditions consistent across models. The comparison therefore focuses on the differences between network architectures rather than changes in learning rate scheduling.

Table 2. Comparison of prediction performance among the evaluated neural models.

Metrics	CNN	U-Net	IU-Net	APU-Net
MSE	0.031752	0.002204	0.000917	0.000451
Dice Coefficient	0.893871	0.923773	0.932113	0.977925
Global Correct	0.903461	0.936549	0.957642	0.994542
SSIM	0.896230	0.934313	0.941202	0.990050
PSNR (dB)	14.98	26.57	30.38	33.46
Average Time (ms)	2.59	1.91	1.97	1.93

4.6. Statistical Significance Analysis

APU-Net and the standard U-Net were each run five times on MNIST to check whether the performance differences were affected by random initialization. The two models were trained and tested separately, following the same experimental protocol. A new set of randomly initialized weights was used at the beginning of each run.

The metrics obtained from these experiments are summarized in **Table 3**. For each metric, the table gives the p-value for the comparison between the two models. It also includes the 95% confidence interval calculated from the five APU-Net runs. All key metrics yield p-values below 0.01. The differences observed between APU-Net and U-Net were therefore statistically significant in these repeated experiments. The confidence intervals for APU-Net were also relatively narrow, in-

dicating that its measured performance varied little across the five initializations. These results suggest that the improvements observed with APU-Net were not limited to a single training run.

Table 3. p-values and 95% confidence intervals for the five-run comparison.

Metrics	p-values	95% Confidence Interval
MSE	<0.01	[0.000448, 0.000473]
Dice Coefficient	<0.01	[0.975164, 0.978544]
Global Correct	<0.01	[0.993388, 0.995113]
SSIM	<0.01	[0.988348, 0.990511]
PSNR	<0.01	[33.25, 33.49]
Average Computation Time	<0.01	[1.905, 2.044]

5. Conclusions

APU-Net is a task-adapted encoder-decoder for predicting the target-plane amplitude distribution A_C from the source-plane phase distribution P_S. Its integrated architecture combines multiscale U-Net feature extraction with normalized convolutional blocks, interpolation-based upsampling, explicit decoder-map alignment, and a capacity-controlled 512-channel bottleneck.

Tests on the MNIST-based dataset give an MSE of 0.000451 and a PSNR of 33.46 dB. Compared with the standard U-Net, the MSE is reduced by 79.5%, while the PSNR increases by 6.89 dB. The comparison with IU-Net also shows lower prediction errors for APU-Net, with little difference in inference time. The timing results show a similar advantage over ASM. At each of the five tested resolutions, APU-Net completes the prediction in less than one-twentieth of the time required by the recorded ASM implementation. This difference is particularly relevant when multiple phase-to-amplitude predictions are needed under the same acoustic configuration.

Additional experiments were conducted using perturbed phase inputs and image patterns from different datasets. The predicted amplitude distributions retained their main spatial features under the tested perturbations. The cross-pattern experiments also provided a means of examining the model's performance on data beyond the MNIST-based training set. These tests offer further evidence of how the model responds to changes in its input.

Despite these promising results, the current method has several limitations. APU-Net is trained on paired data generated under a fixed acoustic configuration, and its applicability to different propagation distances, operating frequencies, transducer geometries, and other acoustic conditions has not yet been systematically established. Future work will focus on extending APU-Net to varying acoustic configurations and improving its generalization across operating conditions.

Data Availability

The datasets generated and/or analyzed during the current study are available

from the corresponding author on reasonable request.

Author Contributions

M.F.: Conceptualization, Investigation, Writing—Original Draft (conceived the research method, conducted experimental validation, and drafted the original manuscript).

X.W.: Methodology, Writing—Review and Editing (developed the research methodology and revised the manuscript).

C.X.: Writing—Review and Editing (participated in manuscript revision and proofreading).

W.Y.: Writing—Review and Editing (participated in manuscript revision and proofreading).

C.W.: Writing—Review and Editing (participated in manuscript revision and proofreading).

All authors: reviewed and approved the final version of the manuscript.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] Yang, L., Jiang, J., Ji, F., Li, Y., Yung, K., Ferreira, A., *et al.* (2024) Machine Learning for Micro- and Nanorobots. *Nature Machine Intelligence*, **6**, 605-618. https://doi.org/10.1038/s42256-024-00859-x

[2] Rufo, J., Cai, F., Friend, J., Wiklund, M. and Huang, T.J. (2022) Acoustofluidics for Biomedical Applications. *Nature Reviews Methods Primers*, **2**, Article No. 30. https://doi.org/10.1038/s43586-022-00109-7

[3] Tian, Z., Yang, S., Huang, P., Wang, Z., Zhang, P., Gu, Y., *et al.* (2019) Wave Number-Spiral Acoustic Tweezers for Dynamic and Reconfigurable Manipulation of Particles and Cells. *Science Advances*, **5**, eaau6062. https://doi.org/10.1126/sciadv.aau6062

[4] André, A.N., Lehmann, O., Govilas, J., Laurent, G.J., Saadana, H., Sandoz, P., *et al.* (2024) Automating Robotic Micro-Assembly of Fluidic Chips and Single Fiber Compression Tests Based-on Θ Visual Measurement with High-Precision Fiducial Markers. *IEEE Transactions on Automation Science and Engineering*, **21**, 353-366. https://doi.org/10.1109/tase.2022.3218686

[5] Zhang, B., Zhu, L., Pan, H. and Cai, L. (2023) Biocompatible Smart Micro/Nanorobots for Active Gastrointestinal Tract Drug Delivery. *Expert Opinion on Drug Delivery*, **20**, 1427-1441. https://doi.org/10.1080/17425247.2023.2270915

[6] Wang, J., Dong, Y., Ma, P., Wang, Y., Zhang, F., Cai, B., *et al.* (2022) Intelligent Micro-/Nanorobots for Cancer Theragnostic. *Advanced Materials*, **34**, Article 2201051. https://doi.org/10.1002/adma.202201051

[7] Rezai, A.R., D'Haese, P., Finomore, V., Carpenter, J., Ranjan, M., Wilhelmsen, K., *et al.* (2024) Ultrasound Blood-Brain Barrier Opening and Aducanumab in Alzheimer's Disease. *New England Journal of Medicine*, **390**, 55-62. https://doi.org/10.1056/nejmoa2308719

[8] Jimenez-Gambin, S., Jimenez, N., Pouliopoulos, A., Benlloch, J.M., Konofagou, E.

and Camarena, F. (2022) Acoustic Holograms for Bilateral Blood-Brain Barrier Opening in a Mouse Model. *IEEE Transactions on Biomedical Engineering*, **69**, 1359-1368. https://doi.org/10.1109/tbme.2021.3115553

[9] Bae, S., Liu, K., Pouliopoulos, A.N., Ji, R. and Konofagou, E.E. (2023) Real-Time Passive Acoustic Mapping with Enhanced Spatial Resolution in Neuronavigation-Guided Focused Ultrasound for Blood-Brain Barrier Opening. *IEEE Transactions on Biomedical Engineering*, **70**, 2874-2885. https://doi.org/10.1109/tbme.2023.3266952

[10] Liu, S., Sun, P., Wang, M., Jiang, Y., Li, J., Jia, Y., *et al.* (2025) Acousto-Dewetting Enables Droplet Microfluidics on Superhydrophilic Surfaces. *Nature Physics*, **21**, 808-816. https://doi.org/10.1038/s41567-025-02844-6

[11] Gu, Y., Chen, C., Rufo, J., Shen, C., Wang, Z., Huang, P., *et al.* (2020) Acoustofluidic Holography for Micro- to Nanoscale Particle Manipulation. *ACS Nano*, **14**, Article 14645. https://doi.org/10.1021/acsnano.0c03754

[12] Ma, Z., Holle, A.W., Melde, K., Qiu, T., Poeppel, K., Kadiri, V.M., *et al.* (2019) Acoustic Holographic Cell Patterning in a Biocompatible Hydrogel. *Advanced Materials*, **32**, Article 1904181. https://doi.org/10.1002/adma.201904181

[13] Bakhtiari-Nejad, M., Elnahhas, A., Hajj, M.R. and Shahab, S. (2018) Acoustic Holograms in Contactless Ultrasonic Power Transfer Systems: Modeling and Experiment. *Journal of Applied Physics*, **124**, Article 244901. https://doi.org/10.1063/1.5048601

[14] Bansal, S., Choi, C., Hardwick, J., Bagchi, B., Tiwari, M.K. and Subramanian, S. (2023) Transmissive Labyrinthine Acoustic Metamaterial-Based Holography for Extraordinary Energy Harvesting. *Advanced Engineering Materials*, **25**, Article 2370017. https://doi.org/10.1002/adem.202370017

[15] Kobayashi, T., Tanaka, T., Yatabe, K. and Oikawa, Y. (2022) Acoustic Application of Phase Reconstruction Algorithms in Optics. 2022 *IEEE International Conference on Acoustics, Speech and Signal Processing* (*ICASSP*), Singapore, 23-27 May 2022, 6212-6216. https://doi.org/10.1109/icassp43922.2022.9747423

[16] Melde, K., Mark, A.G., Qiu, T. and Fischer, P. (2016) Holograms for Acoustics. *Nature*, **537**, 518-522. https://doi.org/10.1038/nature19755

[17] Marzo, A. and Drinkwater, B.W. (2019) Holographic Acoustic Tweezers. *Proceedings of the National Academy of Sciences*, **116**, 84-89. https://doi.org/10.1073/pnas.1813047115

[18] Matsushima, K., Schimmel, H. and Wyrowski, F. (2003) Fast Calculation Method for Optical Diffraction on Tilted Planes by Use of the Angular Spectrum of Plane Waves. *Journal of the Optical Society of America A*, **20**, Article 1755. https://doi.org/10.1364/josaa.20.001755

[19] Zeng, X. and McGough, R.J. (2008) Evaluation of the Angular Spectrum Approach for Simulations of near-Field Pressures. *The Journal of the Acoustical Society of America*, **123**, 68-76. https://doi.org/10.1121/1.2812579

[20] Li, B., Lu, M., Liu, C., Liu, X. and Ta, D. (2022) Acoustic Hologram Reconstruction with Unsupervised Neural Network. *Frontiers in Materials*, **9**, Article 916527. https://doi.org/10.3389/fmats.2022.916527

[21] Lin, Q., Wang, J., Cai, F., Zhang, R., Zhao, D., Xia, X., *et al.* (2021) A Deep Learning Approach for the Fast Generation of Acoustic Holograms. *The Journal of the Acoustical Society of America*, **149**, 2312-2322. https://doi.org/10.1121/10.0003959

[22] Zhong, C., Sun, Z., Li, T., Su, H. and Liu, S. (2023) Real-Time Acoustic Holography with Iterative Unsupervised Learning for Acoustic Robotic Manipulation. 2023 *IEEE International Conference on Robotics and Automation* (*ICRA*), London, 29 May-2

June 2023, 5466-5472. https://doi.org/10.1109/icra48891.2023.10160962

[23] Zhong, C., Jia, Y., Jeong, D.C., Guo, Y. and Liu, S. (2022) AcousNet: A Deep Learning Based Approach to Dynamic 3D Holographic Acoustic Field Generation from Phased Transducer Array. *IEEE Robotics and Automation Letters*, **7**, 666-673. https://doi.org/10.1109/lra.2021.3130368

[24] Ioffe, S. and Szegedy, C. (2015) Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. 32*nd International Conference on Machine Learning*, Lille, 7-9 July 2015, 448-456.

[25] Odena, A., Dumoulin, V. and Olah, C. (2016) Deconvolution and Checkerboard Artifacts. *Distill*, **1**, e3. https://doi.org/10.23915/distill.00003

[26] Ronneberger, O., Fischer, P. and Brox, T. (2015) U-Net: Convolutional Networks for Biomedical Image Segmentation. In: *Lecture Notes in Computer Science*, Springer, 234-241. https://doi.org/10.1007/978-3-319-24574-4_28

[27] Cui, M., Li, Y., Wang, X., Wang, J. and Rong, H. (2024) Rapid Reconstruction of Acoustic Holograms Using Improved IU-Net with Adaptive Learning Rate Scheduling. *IEEE Access*, **12**, 178199-178208. https://doi.org/10.1109/access.2024.3496923