

SoundSil-DS: Deep Denoising and Segmentation of Sound-field Images with Silhouettes

Risako Tanigawa^{1,2}, Kenji Ishikawa¹, Noboru Harada¹, and Yasuhiro Oikawa²

¹NTT Communication Science Laboratories, NTT Corporation, Atsugi, Kanagawa 243-0198, Japan

²Intermedia Art and Science, Waseda University, Shinjuku, Tokyo 169-8555, Japan

{risako.tanigawa, ke.ishikawa, harada.noboru}@ntt.com, yoikawa@waseda.jp

Abstract

Development of optical technology has enabled imaging of two-dimensional (2D) sound fields. This acousto-optic sensing enables understanding of the interaction between sound and objects such as reflection and diffraction. Moreover, it is expected to be used as an advanced measurement technology for sonars in self-driving vehicles and assistive robots. However, the low sound-pressure sensitivity of the acousto-optic sensing results in high intensity of noise on images. Therefore, denoising is an essential task to visualize and analyze the sound fields. In addition to denoising, segmentation of sound and object silhouette is also required to analyze interactions between them. In this paper, we propose sound-field-images-with-object-silhouette denoising and segmentation (SoundSil-DS) that jointly perform denoising and segmentation for sound fields and object silhouettes on a visualized image. We developed a new model based on the current state-of-the-art denoising network. We also created a dataset to train and evaluate the proposed method through acoustic simulation. The proposed method was evaluated using both simulated and measured data. We confirmed that our method can be applied to experimentally measured data. These results suggest that the proposed method may improve the post-processing for sound fields, such as physical model-based three-dimensional reconstruction since it can remove unwanted noise and separate sound fields and other object silhouettes. Our code is available at <https://github.com/nttcsllab/soundsil-ds>.

1. Introduction

Sound is one of the most important cues to understanding scenes as well as vision. For example, self-driving cars and assistive robots are equipped with ultrasonic sonars and microphones, which are used to gather information about their surroundings. Recently, research has been conducted

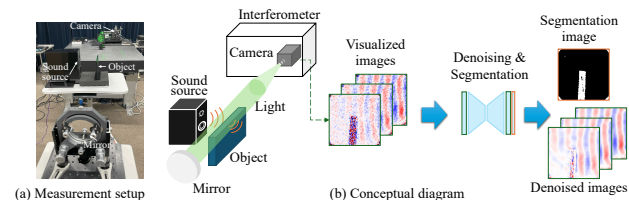


Figure 1. Conceptual diagram of proposed method. (a) Experimental setup for optical sound measurement, which is microphone-free sound measurement device. (b) Conceptual diagram. Sound field with interacting objects is captured as images with high-speed camera. Visualized images are converted to denoised and segmentation images with a DNN.

on converting sound information into vision information and vice versa. Lindell *et al.* have proposed an acoustic non-line-of-sight imaging method for resolving three-dimensional object shapes hidden around corners through acoustic echoes [18]. Davis *et al.* have proposed a method with which sound waves are recovered through object vibrations captured using a high-speed camera [7]. Sheinin *et al.* have proposed a method of sensing sound at high speeds through object-surface vibrations [21]. These studies demonstrated the potential of capturing sound as images, paving the way for further advancements in the field.

A sound-visualization technique involving directly capturing and visualizing the density variations in air caused by sound has been proposed [28]. Such acousto-optic sensing can capture sound without any microphones by observing modulations of the phase of light passing through sound fields. By using high-speed cameras as sensors, it becomes possible to create visual representations of invisible sound waves as images [15, 23]. However, a significant challenge remains: the phase modulation of light induced by sound is extremely small, leading to a high level of noise in the measured images. Ishikawa *et al.* [14] have proposed deep sound-field denoiser (DSFD), the first denoiser to use deep neural networks (DNNs) for noise reduction in sound-field images, demonstrating that DNN-based meth-

ods achieve superior denoising performance compared with conventional filtering methods.

With the optical technology and signal processing method, we can understand the nature of sound propagation. Therefore, the interaction between sound and objects, such as reflections and diffractions, can be visualized using acousto-optic sensing. To analyze such interactions between sound and objects, both denoising and segmentation of the sound field and object regions on visualized images should be done simultaneously.

We propose a method for simultaneous denoising and segmentation of sound-field images including object silhouettes. A conceptual diagram is presented in Fig. 1. With this method, optically measured sound-field images with object silhouettes are denoised and segmented using a DNN. The sound field on the laser path is captured with a high-speed camera. The area where the object blocks the laser light can be visualized as noisy silhouettes in the visualized images. The visualized images are converted to denoised and segmentation images with the DNN. The DNN is constructed based on the state-of-the-art (SOTA) denoising network, which also has the potential for per-pixel feature segmentation. We created a dataset with acoustic simulation since there is no dataset that includes acoustic scattering by objects. Denoising and segmentation are expected to (1) enable analysis of the propagation, reflection, and diffraction of sound waves in space and (2) be used as an advanced measurement technology for sonars in self-driving vehicles and assistive robots. The contributions of our work are summarized as follows:

- We propose a method for simultaneous denoising and segmentation of sound-field images with silhouettes.
- We created a dataset considering acoustic scattering caused by various shapes of objects.
- We confirmed that the proposed method performed effectively on denoising and segmentation tasks.

2. Related Work

2.1. Acousto-optic Sensing

Acousto-optic sensing detects sounds by illuminating the sound fields with laser light from a distance, by using the changes in the refractive index of air due to density variations caused by sound. Since the method can capture sound without microphones, it can be useful to measure sound where microphone cannot be used such as narrow spaces and inside airflow. This technique has been applied to various situations such as visualizing sound field generated from loudspeakers [11, 15, 20], measuring flow-induced sounds [22, 25], and microphone calibrations [12, 16]. Using a high-speed camera to capture sound-field images provides an intuitive understanding of acoustic phenomena [12, 15, 25].

While they can non-intrusively capture 2D sound fields, the measured data often contain a significant amount of noise due to the small modulation of the light phase. An example of the measured data of visualized images is illustrated in Fig. 1. The sound wave propagates from right to left within the images, with reflections and diffractions occurring at the reflector. The data contain a significant noise; therefore, the diffracted waves are almost invisible.

2.2. Sound-field-image Denoising

Sound-field images captured by a high-speed camera are three-dimensional data in two dimensions of space and one dimension of time. Sound-field-image denoising has been done for both space and time dimensions.

Classical Filters For more than a decade, classical filters have been used to reduce noise in sound fields [6, 24, 30]. The most straightforward method involves time-domain filtering [30]. The time-directional signals in each pixel can be considered similar to microphone signals where 2D sound fields are captured by high-speed camera. Hence, by applying time-domain filtering on a per-pixel basis, it becomes feasible to extract images corresponding to specific frequencies. Although the time-domain filters can remove noise with frequencies other than that of sound, noise components that fluctuate at the same frequency as the sound cannot be removed. In contrast, by using spatio-temporal filters [6, 24], we can extract components that satisfy the frequency of sound in both the time and spatial domains.

Deep Sound-Field Denoiser (DSFD) [14] The DNN-based sound-field denoising method DSFD has been proposed. It is the first attempt at sound-field denoising using a DNN. The network of DSFD is based on nonlinear activation free network (NAFNet) [3], which is a network for natural image denoising. The input data consists of frequency-domain data obtained by applying Fourier transform (FT) in the time direction to each pixel of a sound-field video. The real and imaginary parts of the complex amplitudes at a specific frequency are treated as two separate images, which are then stacked along the channel direction, resulting in input data $X \in \mathbb{R}^{N \times C \times W \times H}$, where N is the number of images, $C = 2$ is the number of channels, and W and H are the number of pixels in width and height. The training data has been generated from 2D sound-field simulations in the frequency domain. Supervised learning is carried out using the simulated noisy images as inputs. By using DNNs, denoising can be achieved with higher accuracy compared with traditional classical filtering methods.

2.3. Image Denoising and Segmentation

As demonstrated with DSFD, 2D sound-field data can be regarded as images, allowing the application of DNNs com-

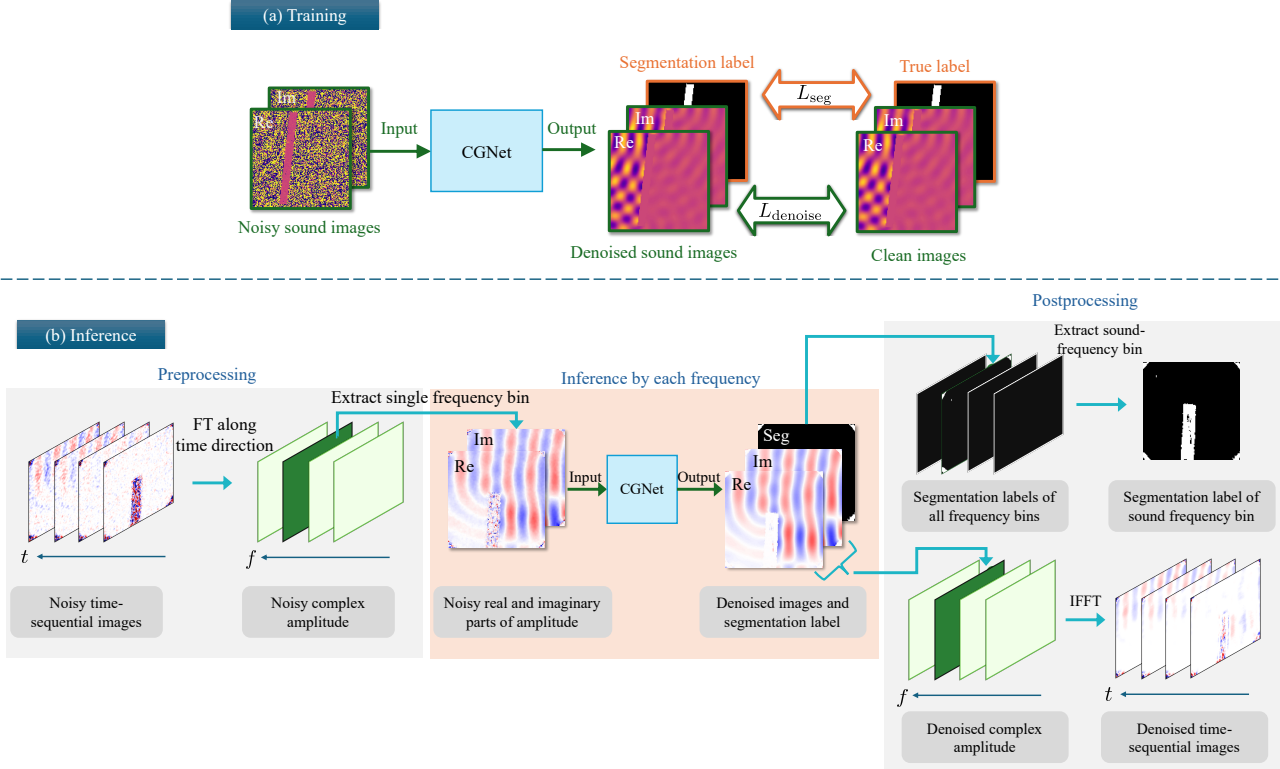


Figure 2. Overview of our approach. (a) Training process. Two channels of noisy sound images are input into network. Output images have three channels: first two channels are for denoising and last one channel is for segmentation. Loss of denoised and segmented images is calculated separately for each ground truth image. (b) Inference process. Experimentally measured time sequential images are converted to frequency domain by Fourier transform (FT). Each frequency complex amplitude is turned into real and imaginary images and input into trained model. Denoised images of all frequency bins are converted to time-sequential images by inverse FT. Segmentation image at sound frequency is extracted as final segmentation label.

monly used for RGB/grayscale images. Numerous natural-image denoising models have been proposed, including convolutional neural network (CNN)-based [3, 9, 10, 32, 33] and transformer-based models [29, 31]. Denoising models have been proposed that use lightweight CNNs instead of transformers. In fact, the top three SOTA denoising models on the Smartphone Image Denoising Dataset (SIDD) [1] have used CNNs [3, 9, 33]. These CNN-based models are designed to retain global information, achieving impressive accuracy. For instance, Cascaded Gaze Net (CGNet) [9], which outperforms the NAFNet [3], has achieved a peak signal-to-noise ratio (PSNR) of 40.39 dB on the SIDD. Therefore, we believed it better to use this SOTA network as a base network other than NAFNet in our study.

The task of joint denoising and segmentation, which is the objective of this study, is explored in the field of microscopy imaging. DenoiSeg [2], for instance, is a method that performs both denoising and segmentation on biological images. DenoiSeg achieves this multitasking by adding channels corresponding to the segmentation classes to the decoder of a denoising network in Noise2Void [17]. Losses are then calculated for each task, enabling the model to pro-

duce both denoised and segmentation images. The largest difference of the proposed method is using self-supervised learning due to the difficulty in obtaining clear images. Sound-field images can be generated from acoustic simulations, making supervised learning a rational approach.

3. Proposed Methods

To simultaneously carry out denoising and segmentation, we use the SOTA denoising network, CGNet [9], as a base network. We increase the number of output channels in the final layer of CGNet to three channels. The first two channels are for denoising sound images, that same as with DSFD. The last channel is for the segmentation of object silhouettes. By increasing the number of channels in the final layer, the proposed method can accomplish both tasks, resulting in lower computational costs compared with training denoising and segmentation sequentially.

3.1. Proposed Architecture

Overviews of training and inference processes are shown in Fig. 2(a) and Fig. 2(b), respectively. The same as with

DSFD, denoising is carried out in the frequency domain. To achieve this, frequency-domain data, obtained by applying a 1D FT in the time direction to time-domain sound-field videos are used. In the training process (Fig. 2(a)), simulated sound-field images with added noise are input to CGNet. The input channels consist of real and imaginary parts of the frequency-domain data obtained from 1D FT. The number of channels of the last layer of the network is set to three channels: first two channels are for denoising $\hat{X}_{\text{denoise}} \in \mathbb{R}^{N \times 2 \times W \times H}$ and last channel is for segmentation $\hat{X}_{\text{seg}} \in \mathbb{R}^{N \times 1 \times W \times H}$. The range of the segmentation data is converted to $\hat{X}'_{\text{seg}} \in [0, 1]$ using a sigmoid function. Segmentation labels are the binary labels obtained by the thresholding of \hat{X}'_{seg} . The binary labels are 0 for sound and 1 for silhouette class. The loss function is the weighted sum of L_{denoise} and L_{seg} as

$$L = L_{\text{denoise}} + \lambda L_{\text{seg}}, \quad (1)$$

where λ is the weighting coefficient that balances the two losses. For denoising, the negative PSNR loss function is used as well as that of CGNet. For segmentation, a combination of binary cross entropy loss and dice loss is used to reduce the bias in the number of pixels in the sound and silhouette classes:

$$L_{\text{seg}} = (1 - \alpha) L_{\text{BCE}} + \alpha L_{\text{Dice}}, \quad (2)$$

where α is the weighting coefficient. These loss functions are determined based on the result of the preliminary experiment (See details in the supplementary material).

The inference process is shown in Fig. 2(b). To carry out inference in the frequency domain, noisy time-sequential images $x_{\text{raw}} \in \mathbb{R}^{W \times H \times T}$ are converted to noisy complex amplitude $X_{\text{raw}} \in \mathbb{C}^{W \times H \times F}$ with 1D FT along the temporal axis, where T and F are the number of data samples in temporal and frequency axes, respectively. Denoising and segmentation are carried out for each frequency bin of X_{raw} . The image in which the real and imaginary parts of the i -th frequency bin $X_{\text{raw},i}$ are arranged in the channel direction is input to the network in the same manner as in the training process. After carrying out inference for all frequency bins, denoised images are converted to time-sequential data with inverse FT. The segmentation image at the sound frequency is the final segmentation label since the image at this sound frequency exhibits the highest contrast between the sound field and silhouettes and is considered to be easier to separate those two classes.

3.2. Dataset Creation

Since no dataset exists for training the network, we created a dataset of sound fields including objects. We used acoustic numerical simulation to create a training dataset because it is difficult to collect sound-field data under various conditions through experiments.

Simulation Conditions To simulate sound fields including objects, we conducted a simulation of the sound fields in the time domain with MATLAB [13] using the k-Wave toolbox [27]. The simulation setup is shown in Fig. 3(a). Following DSFD, to obtain the image size of 128×128 pixels, the observation area was set to $1.28 \text{ m} \times 1.28 \text{ m}$, and observation points were set in a grid pattern at intervals of 0.01 m . Sound sources were randomly placed outside the observation area, within a range of $2.56 \text{ m} \times 2.56 \text{ m}$. Objects were set inside the observation area. The shapes of objects included ellipses, lines, and polygons, and the parameters positions were randomly selected. The medium other than the object was air, and the object was made of expanded polystyrene (EPS) to enable stable calculations. Thus, the reflectivity of the object was 93.2%.

The simulation conditions are listed in Tab. 1. The frequency range of sound f_s was set to $90 \leq f_s \leq 2800 \text{ Hz}$, which corresponds to the wavenumber k being $1.66 \leq k \leq 51.7 \text{ rad/m}$. The sound frequency for each condition was fixed, which means each sound source had the same frequency and was randomly selected from a uniform distribution. The amplitude of the first sound source was set to 1 Pa , and those of the other sound sources were randomly selected from a uniform distribution between 0.1 to 1 Pa . Calculated time-sequential data were converted to the frequency domain with FT and the data at the sound frequency were extracted. The total amount of images was $55,000$, with $11,000$ images for each number of sound sources. Sample simulated data are shown in the second row of Fig. 3(b). Reflections and diffractions were simulated to occur around the silhouettes highlighted in white in the first row of Fig. 3(b).

Noisy-Data Creation Since the noise characteristics between sound-field regions and object-silhouette regions can be different, we added different types of noise. For the sound-field regions, we added white noise with different SNRs. The SNRs were randomly selected from a uniform distribution between -20 to 20 dB . For the silhouette regions, we created noise on the basis of experimentally obtained data. The details of the noisy data creation for silhouette regions are in the supplementary material. We estimated the probability density function (PDF) from empirical data using kernel density estimation [5]. On the basis of the estimated PDF, we generated noise for silhouette regions by using the inverse transform sampling method [26]. The SNRs of the noise for silhouette regions were also selected from a uniform distribution between -20 to 20 dB . However, since clean-image values in silhouette regions are zero, SNRs were set for signals in the sound-field regions.

Created noisy data are shown in the last row of Fig. 3(b). Because the SNRs of the silhouette and sound-field regions were set independently, the intensities of the noise differed.

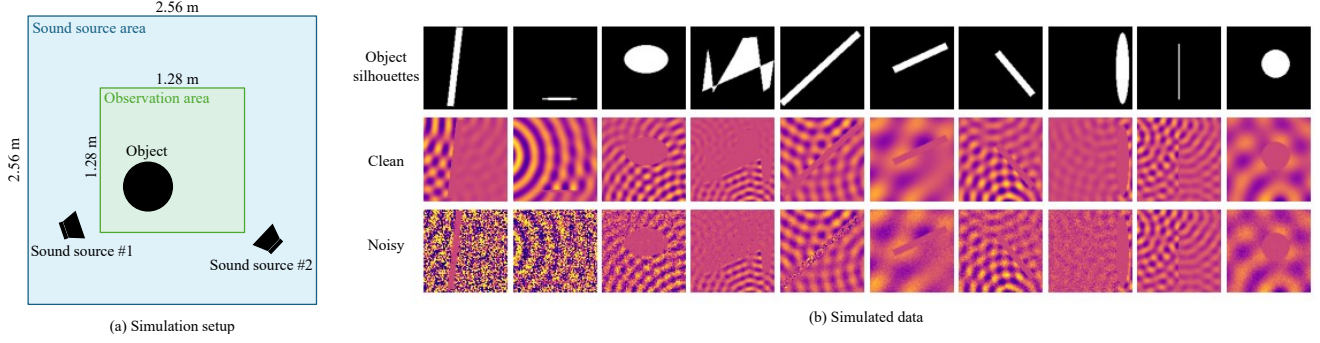


Figure 3. Dataset creation. (a) Simulation setup. Sound sources are installed outside observation area. Objects are installed inside observation area. (b) Simulated data. Top row shows object silhouettes, second row shows clean simulated images, and bottom row shows noisy images with noise added to clean images. Color indicates real part of complex amplitude, ranging from -1.0 to 1.0 .

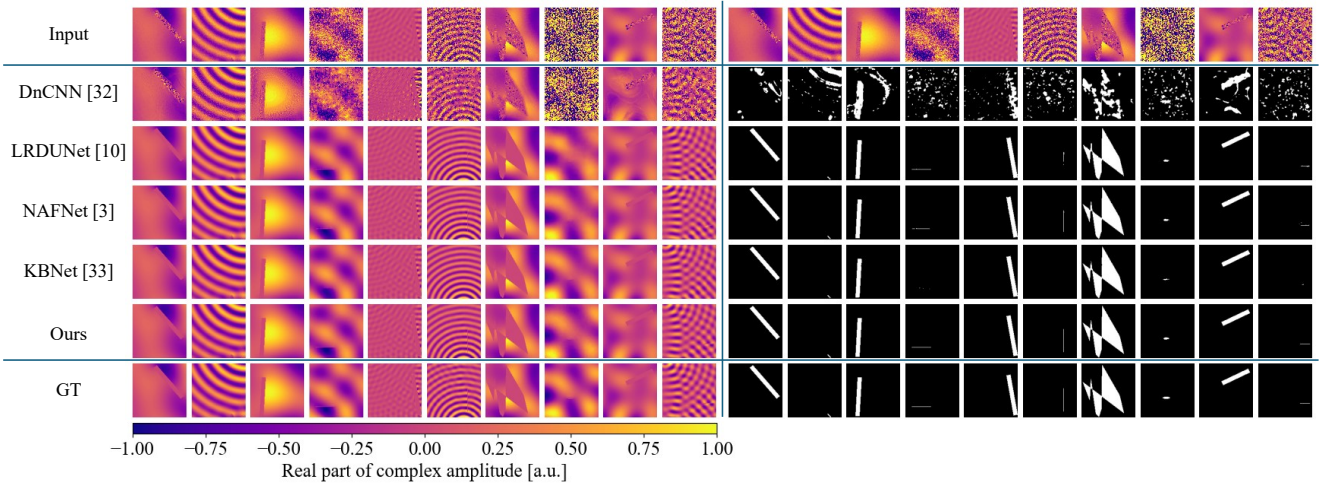


Figure 4. Qualitative results. Top row shows input images. From second to sixth rows, denoised and segmented images are shown. Last row shows GT images. Left ten columns are for denoising and right ten columns are for segmentation. Input images on right ten columns are same as those in left ten columns. Color indicates real part of complex amplitude.

4. Experiment

4.1. Implementation Details

The network was implemented by PyTorch on the basis of the official implementation of CGNet [8, 9]. The CGNet consists of an input layer, four encoder blocks, one middle block, four decoder blocks, and an output layer. The numbers of encoder blocks were set to 2, 2, 4, and 6, respectively, in order of increasing depth. The number of middle blocks was 10. The number of decoder blocks were set to 2, 2, 2, and 2, respectively, in order of decreasing depth.

The input image size was set to 128×128 pixels, and the number of channel was two. The weighting coefficient λ of the loss function in Eq. (1) was set to $\lambda = 10$ on the basis of experiments conducted with a small amount of data. The α of the loss function in Eq. (2) was set to $\alpha = 0.5$. The AdamW optimizer was used where the learning rate was

0.001, weight decay was 0.0, and β_1 and β_2 were 0.9 and 0.9, respectively. The cosine annealing scheduler was used where the maximum number of iterations was 400,000, and the minimum learning rate was $1e-7$. The batch size was 16, and the number of epochs was 20. The training time was approximately 3.5 hours with a single NVIDIA GeForce RTX 4090 GPU. The number of training, validation, and evaluation images were 50000, 2500, and 2500, respectively.

The PSNR and structural similarity (SSIM) were used for denoising, and intersection over union (IoU) was used for segmentation as the evaluation metrics. The IoUs were calculated for class 1, i.e., the silhouette regions.

4.2. Compared Models

In our proposed method, we used CGNet as the base network. To validate the use of CGNet, we also evaluated the performance when using existing networks as the base

Parameter	Values
Spatial grid size [m]	0.01
Temporal discretization step [s]	1.21×10^{-5}
Speed of sound in air [m/s]	340
Speed of sound in EPS [m/s] [4]	414
Density of air [kg/m ³]	1.21
Density of EPS [kg/m ³] [4]	28.0
Number of sound sources S	$S \in \{1, 2, 3, 4, 5\}$
Frequency of sound f_s [Hz]	$90 \leq f_s \leq 2800$
Amplitude of sound source p_s [Pa]	$0.10 \leq p_s \leq 1.0$

Table 1. Simulation conditions.

Base Network	PSNR [dB]	SSIM	IoU
DnCNN [32]	16.6	0.603	0.284
LRDUNet [10]	39.6	0.976	0.970
NAFNet [3]	40.8	0.983	0.977
KBNet [33]	42.0	0.986	0.976
CGNet [9] (Ours)	43.2	0.987	0.986

Table 2. Quantitative results. Ours performed best.

network. Compared networks were selected from denoising networks since denoising is more complicated task than segmentation, which is a two class classification in our task.

The following four conventional denoising networks were used: denoising convolutional neural network (DnCNN) [32], lightweight residual dense neural network based on the U-net neural network (LRDUNet) [10], NAFNet [3], and kernel basis network (KBNet) [33]. Since all those four networks were for denoising, we changed the number of output channels of the last layer to be the same as that of the proposed method. Under the training of DnCNN, LRDUNet, NAFNet, and KBNet, the λ of the loss function in Eq. (1) were set to 0.001, 0.01, 0.005, and 0.01, respectively. The batch sizes for DnCNN, LRDUNet, NAFNet, and KBNet were set to 32, 32, 32, and 16, respectively. The number of epochs for all models was 20. The implementation details are in the supplementary material.

4.3. Results

The results are shown in Tab. 2 and Fig. 4. The table shows that our method, which used CGNet as a base network, recorded the highest scores in terms of PSNR, SSIM, and IoU. As shown in Fig. 4, both sound-field and silhouette regions were denoised except for DnCNN. In the segmentation results, small objects that could not be detected by KBNet are successfully detected by ours. From these results,

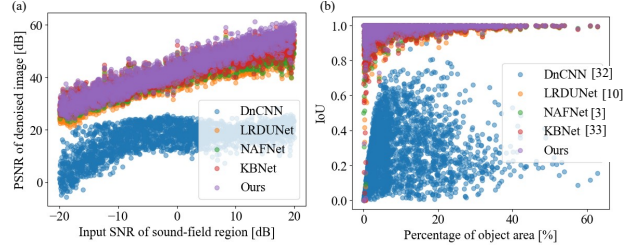


Figure 5. Analyses of results. (a) PSNRs of denoised images relative to SNRs of sound field in input images. PSNRs improved as input SNR increased except for DnCNN. (b) IoUs of segmentation images relative to percentage of object silhouettes' area. IoUs tended to decrease where areas were small.

we confirmed that using CGNet as a base network was valid for sound-field-image joint denoising and segmentation.

To analyze denoising ability, we investigated the effect of the input SNRs of sound-field regions on the PSNRs, as plotted in Fig. 5(a). The horizontal axis means the input SNRs for sound-field regions, and the vertical axis means the output PSNRs. Except for DnCNN (blue dots), the output PSNRs positively correlated with the input SNRs for sound-field regions. Although the output PSNRs performed similarly where the input SNRs were low, except for DnCNN, ours (purple dots) showed improved output PSNRs at high input SNRs around 10 to 20 dB.

To analyze segmentation ability, we investigated the effect of the percentages of silhouette areas on the IoUs as plotted in Fig. 5(b). The percentages of silhouette areas were calculated as the number of pixels in the object area divided by the total number of pixels in the image. Except for DnCNN (blue dots), the number of data points with low IoUs tended to increase as the size of the object area decreased. Ours (purple dots) showed that there are many data points near 1.0 IoU even where the percentages of silhouette areas were small around 0 to 10%.

5. Evaluation for Experimental Data

To confirm the applicability to the experimentally measured data, we applied our method to the following two types of experimental data: (1) A sound field diffracted with a thin plate, and (2) a sound field generated with a wooden finger castanets, a percussive musical instrument. All the data were captured by parallel phase-shifting interferometry (PPSI) [15], which is often used for sound-field imaging due to its high sensitivity and spatial resolution. PPSI can capture sound-field images within 100 mm in diameter by using a high-speed polarization camera [19].

5.1. Sound Field with Reflection and Diffraction

Experimental Setup We recorded a sound field with reflection and diffraction where a reflector was set in front of

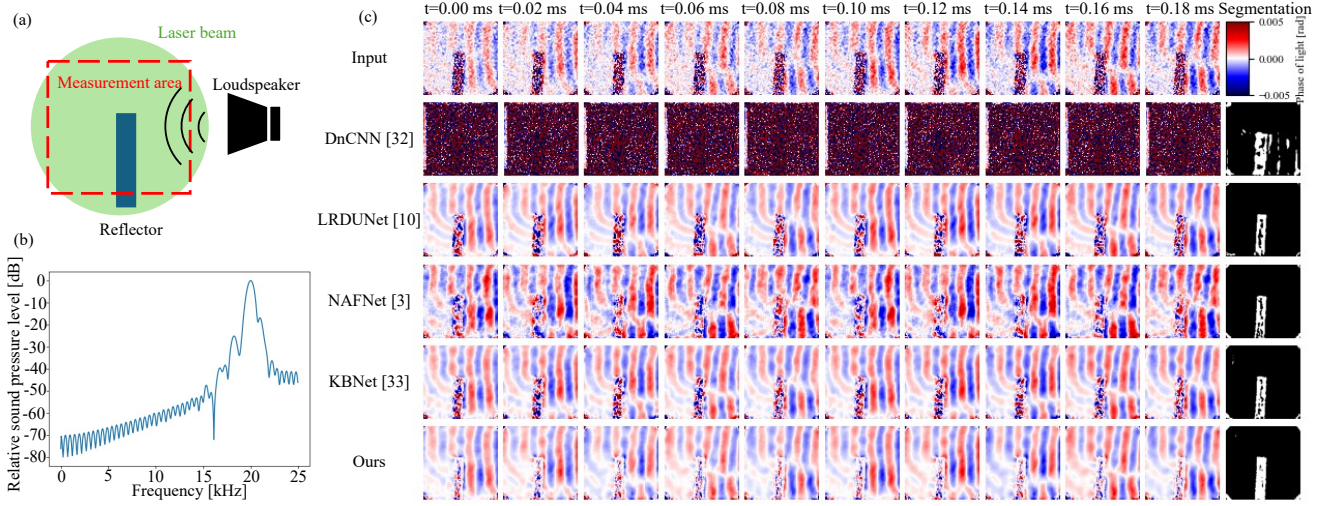


Figure 6. Experimental results of sound field with reflection and diffraction. (a) Experimental setup. Reflector was installed 15 cm from loudspeaker. (b) Frequency spectrum of input data. (c) Experimental results of denoising and segmentation. Top row shows input images in time domain and other rows show denoised and segmentation results. Color indicates phase of light detected with PPSI.

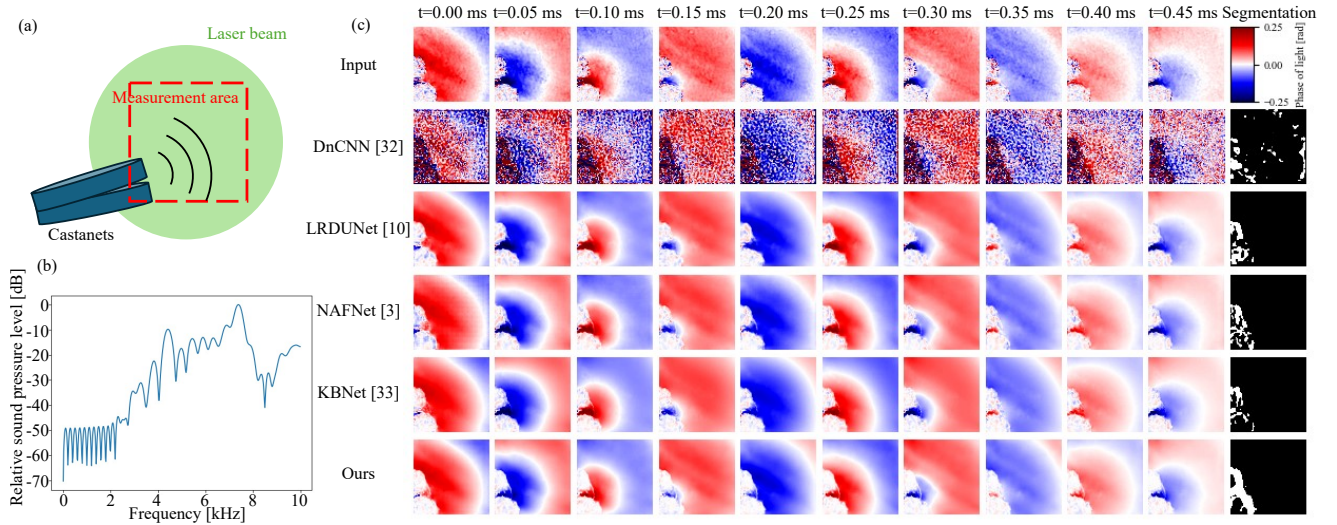


Figure 7. Experimental results of castanets sound. (a) Experimental setup. Castanets was located at bottom-left of imaging area. (b) Frequency spectrum of input data. (c) Experimental results of denoising and segmentation. Top row shows input images in time domain and other rows show denoised and segmentation results. Color indicates phase of light detected with PPSI.

a loudspeaker. The schematic diagram of the setup is shown in Fig. 6(a). The loudspeaker (NS-BP200, YAMAHA) was set outside the measurement area. The reflector was set inside the measurement area and 150 mm from the loudspeaker. The dimensions of the reflector were 150 mm in height, 10 mm in width, and 200 mm in depth. A 20-kHz sinusoidal wave was emitted from the loudspeaker. The frame rate of the camera was set to 50000 frames per second. A frequency spectrum of captured data was shown in Fig. 6(b). The spectrum was calculated from a time series signal at the index of $[20, 100]$, where $[\cdot, \cdot]$ represents

the pixel coordinates in terms of height and width, respectively. Since the sound emitted from the loudspeaker was a 20-kHz sinusoidal wave, the spectrum exhibited a distinct peak at 20 kHz.

Results The results are shown in Fig. 6(c). The top row is the input data from $t = 0.00$ to $t = 0.18$ ms with interval of 0.02 ms. The sound wave propagated from right to left of the imaging area. The reflection and diffraction occurred around the reflector. Since the color range was kept consistent across all conditions to ensure fair visualization,

the denoised results of DnCNN were saturated; therefore, it performed the worst. Although the denoised results of LRDUNet seemed to clearly visualize the sound wave, the amplitudes of the diffracted waves seemed larger than in the input data. Ours showed that noise was effectively removed while maintaining amplitudes of sound waves close to the input data. Ours also excelled in denoising silhouette regions compared with the others. The time-series denoised video is provided as supplementary material. In terms of segmentation, although there were undetected pixels, ours could estimate the least number of undetected pixels.

5.2. Sound Field with Sound Source Object

Experimental Setup We also conducted denoising and segmentation for the sound field including the sound source object. We used wooden finger castanets, a percussive musical instrument as a sound source. The schematic diagram of the experimental setup is shown in Fig. 7(a). The castanets was installed, the edge of which was included in the measurement area. The castanets was played by human hand, and the sound was recorded with a PPSI system at 20,000 frames per second. A frequency spectrum of captured data was shown in Fig. 7(b). The spectrum was calculated from a time series signal at the index of [90, 40]. The spectrum exhibited a broad frequency distribution, indicating that the signal contained multiple frequencies.

Results The results are shown in Fig. 7(c). The top row is the input data from $t = 0.00$ to $t = 0.45$ ms with intervals of 0.05 ms. The sound-wave propagation from castanets, located at bottom-left corner, can be seen. Since the color range was consistent across all conditions, the denoised results of DnCNN were saturated. All networks except DnCNN could denoise the fine noise, and the sound wave was smoothed. By KBNNet, the wavefront's shape was more rounded, which was observed especially at $t = 0.10$ and 0.30 ms, whereas ours kept the shape of the wavefront. The time-series denoised video is provided as supplementary material. Although segmentation did not work well in all models, ours could capture the edge of the castanets. Post-processing, such as dilation, would fill the holes.

6. Comparison of Single-tasking and Multitasking

We evaluated how carrying out denoising and segmentation at the same time would change the accuracy compared with carrying out each as a single task. To do so, denoising and segmentation were conducted separately on the basis of CGNet. The results are shown in Tab. 3. The performance of denoising and segmentation by multitasking was slightly better than those of single-tasking. We also confirmed that multitasking can be implemented with minimal

Task	PSNR [dB]	SSIM	IoU	Inference time [ms]	Model size [MB]
Denoising	43.1	0.986	-	21.99	1243.87
Segmentation	-	-	0.984	21.82	1243.74
Ours	43.2	0.987	0.986	22.10	1244.01

Table 3. Comparison of single-tasking and multitasking. Multitasking showed slightly better performance than single-tasking without a significant increase in inference time and model size.

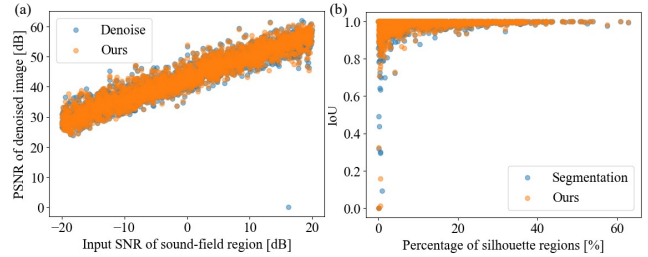


Figure 8. Comparison of single-tasking and multitasking results. (a) PSNRs of denoised images relative to input SNRs of sound region. Blue and orange dots show results of single-tasking and multitasking, respectively. No significant difference in denoising performance between single-tasking and multitasking were observed. (b) IoUs of segmentation images relative to percentage of object silhouettes' area. Multitasking IoUs (Ours) were higher where silhouette regions were small.

impact on inference time of single image and total model size for single-image input.

For further investigation, PSNRs relative to input SNRs and IoUs relative to the percentage of object silhouettes' region are shown in Fig. 8. From Fig. 8(a), there was no significant difference in denoising performance between single-tasking and multitasking. On the other hand, according to Fig. 8(b), multitasking IoUs (blue dots) tended to be higher than single-tasking IoUs (orange dots) where object silhouettes' areas were less than 10 %.

7. Conclusions

We proposed a denoising and segmentation method for 2D sound-field images with object silhouettes. To handle the sound fields with acoustic scattering by objects, we created a dataset through acoustic simulation. Multitasking was realized by using the output of the final layer of CGNet as channels for denoising and segmentation and calculating the loss function for each task. We confirmed that the proposed method can be applied to both simulated and experimental data. We believe that this method can be used for analyzing sound fields with interacting objects and for sonars in self-driving vehicles and assistive robots. Future work includes improving segmentation performance for experimental data, evaluating with recent segmentation architectures, and enabling moving-object segmentation.

References

- [1] Abdelrahman Abdelhamed, Stephen Lin, and Michael S. Brown. A high-quality denoising dataset for smartphone cameras. In *CVPR*, 2018. [3](#)
- [2] Tim-Oliver Buchholz, Mangal Prakash, Alexander Krull, and Florian Jug. DenoiSeg: Joint denoising and segmentation. pages 324–337, 2020. [3](#)
- [3] Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. In *ECCV*, pages 17–33, 2022. [2](#), [3](#), [6](#)
- [4] Wensu Chen, Hong Hao, Dylan Hughes, Yanchao Shi, Jian Cui, and Zhong-Xian Li. Static and dynamic mechanical properties of expanded polystyrene. *Materials & Design*, 69:170–180, 2015. [6](#)
- [5] Yen-Chi Chen. A tutorial on kernel density estimation and recent advances. *Biostatistics & Epidemiology*, 1(1):161–187, 2017. [4](#)
- [6] Nachanant Chitanont, Kohei Yatabe, Kenji Ishikawa, and Yasuhiro Oikawa. Spatio-temporal filter bank for visualizing audible sound field by schlieren method. *Applied Acoustics*, 115:109–120, 2017. [2](#)
- [7] Abe Davis, Michael Rubinstein, Neal Wadhwa, Gautham Mysore, Frédo Durand, and William T. Freeman. The visual microphone: Passive recovery of sound from video. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 33(4):79:1–79:10, 2014. [1](#)
- [8] Amirhosein Ghasemabadi, Muhammad Kamran Janjua, Mohammad Salameh, Chunhua Zhou, Fengyu Sun, and Di Niu. CascadedGaze. <https://github.com/Ascend-Research/CascadedGaze>, 2024. Accessed on 1st, July, 2024. [5](#)
- [9] Amirhosein Ghasemabadi, Muhammad Kamran Janjua, Mohammad Salameh, Chunhua Zhou, Fengyu Sun, and Di Niu. CascadedGaze: Efficiency in global context extraction for image restoration. *Transactions on Machine Learning Research*, page 20 pages, 2024. [3](#), [5](#), [6](#)
- [10] Javier Gurrola-Ramos, Oscar Dalmau, and Teresa Alarcón. U-Net based neural network for fringe pattern denoising. *Optics and Lasers in Engineering*, 149:14 pages, 2022. [3](#), [6](#)
- [11] Michael J. Hargather, Gary S. Settles, and Matthew J. Madalis. Schlieren imaging of loud sounds and weak shock waves in air near the limit of visibility. *Shock Waves*, 20:9–17, 2010. [2](#)
- [12] Denny Hermawanto, Kenji Ishikawa, Kohei Yatabe, and Yasuhiro Oikawa. Determination of microphone acoustic center from sound field projection measured by optical interferometry. *The Journal of the Acoustical Society of America*, 153(2):1138–1146, 2023. [2](#)
- [13] The MathWorks Inc. MATLAB version: 24.1.0.2603908 (R2024a). [4](#)
- [14] Kenji Ishikawa, Daiki Takeuchi, Noboru Harada, and Takehiro Moriya. Deep sound-field denoiser: optically-measured sound-field denoising using deep neural network. *Optics Express*, 31:33405–33420, 2023. [1](#), [2](#)
- [15] Kenji Ishikawa, Kohei Yatabe, Nachanant Chitanont, Yusuke Ikeda, Yasuhiro Oikawa, Takashi Onuma, Hayato Niwa, and Minoru Yoshii. High-speed imaging of sound using parallel phase-shifting interferometry. *Optics Express*, 24(12):12922–12932, Jun 2016. [1](#), [2](#), [6](#)
- [16] Triantafillos Koukoulas and Ben Piper. Towards direct realisation of the SI unit of sound pressure in the audible hearing range based on optical free-field acoustic particle measurements. *Applied Physics Letters*, 106(16):164101, 2015. [2](#)
- [17] Alexander Krull, Tim-Oliver Buchholz, and Florian Jug. Noise2Void - learning denoising from single noisy images. In *CVPR*, 2019. [3](#)
- [18] David B. Lindell, Gordon Wetzstein, and Vladlen Koltun. Acoustic non-line-of-sight imaging. In *CVPR*, June 2019. [1](#)
- [19] Takashi Onuma and Yukitoshi Otani. A development of two-dimensional birefringence distribution measurement system with a sampling rate of 1.3MHz. *Optics Communications*, 315:69–73, 2014. [6](#)
- [20] Tatsuya Sakoda and Yoshito Sonoda. Visualization of sound field with uniform phase distribution using laser beam microphone coupled with computerized tomography method. *Acoustical Science and Technology*, 29(4):295–299, 2008. [2](#)
- [21] Mark Sheinin, Dorian Chan, Matthew O’Toole, and Srinivasa G. Narasimhan. Dual-shutter optical vibration sensing. In *CVPR*, 2022. [1](#)
- [22] Yoshito Sonoda and Yoichi Nakazono. Development of optophone with no diaphragm and application to sound measurement in jet flow. *Advances in Acoustics and Vibration*, 2012(1):17 pages, 2012. [2](#)
- [23] Yuki Takase, Kazuki Shimizu, Shogo Mochida, Tomoyoshi Inoue, Kenzo Nishio, Sudheesh K. Rajput, Osamu Matoba, Peng Xia, , and Yasuhiro Awatsuji. High-speed imaging of the sound field by parallel phase-shifting digital holography. *Applied Optics*, 60(4):A179–A187, 2021. [1](#)
- [24] Risako Tanigawa, Kohei Yatabe, and Yasuhiro Oikawa. Guided-spatio-temporal filtering for extracting sound from optically measured images containing occluding objects. In *ICASSP*, pages 945–949, 2019. [2](#)
- [25] Risako Tanigawa, Kohei Yatabe, and Yasuhiro Oikawa. Experimental visualization of aerodynamic sound sources using parallel phase-shifting interferometry. *Experiments in Fluids*, 61(206):10 pages, 2020. [2](#)
- [26] Roberto Tempo, Giuseppe Calafiore, and Fabrizio Dabbene. *Random Number and Variate Generation*, pages 193–215. Springer London, London, 2013. [4](#)
- [27] Bradley E. Treeby and Benjamin T. Cox. k-Wave: MATLAB toolbox for the simulation and reconstruction of photoacoustic wave fields. *Journal of Biomedical Optics*, 15:12 pages, 2010. [4](#)
- [28] Samuel A. Verburg, Kenji Ishikawa, Efren Fernandez-Grande, and Yasuhiro Oikawa. A century of acousto-optics-from early discoveries to modern sensing of sound with light. *Acoustics Today*, 19(3):54–62, 2023. [1](#)
- [29] Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. Uformer: A general u-shaped transformer for image restoration. In *CVPR*, pages 17683–17693, June 2022. [3](#)
- [30] Kohei Yatabe, Risako Tanigawa, Kenji Ishikawa, and Yasuhiro Oikawa. Time-directional filtering of wrapped phase

- for observing transient phenomena with parallel phase-shifting interferometry. *Optics Express*, 26:13705–13720, 2018. 2
- [31] Syed Waqas Zamir, Aditya Arora, Salman Hameed Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *CVPR*, pages 5728–5739, June 2022. 3
- [32] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017. 3, 6
- [33] Yi Zhang, Dasong Li, Xiaoyu Shi, Dailan He, Kangning Song, Xiaogang Wang, Honwei Qin, and Hongsheng Li. KBNNet: Kernel basis network for image restoration. *arXiv preprint arXiv:2303.02881*, 2023. 3, 6