Supplemental Document

optica

Exceeding the limits of 3D fluorescence microscopy using a dual-stage-processing network: supplement

HAO ZHANG,[1,†] YUXUAN ZHAO,[1,†] CHUNYU FANG,[1] GUO LI,[1] MENG ZHANG,[2] YU-HUI ZHANG,[2,3] AND PENG FEI[1,*]

[1] *School of Optical and Electronic Information-Wuhan National Laboratory for Optoelectronics, Huazhong University of Science and Technology, Wuhan 430074, China*
[2] *Britton Chance Center for Biomedical Photonics, Wuhan National Laboratory for Optoelectronics, Huazhong University of Science and Technology, Wuhan 430074, China*
[3] *MoE Key Laboratory for Biomedical Photonics, Huazhong University of Science and Technology, Wuhan 430074, China*
[†] *These authors contributed equally to this work.*
[*] *Corresponding author: feipeng@hust.edu.cn*

Supplement DOI: https://doi.org/10.6084/m9.figshare.13114460

Parent Article DOI: https://doi.org/10.1364/OPTICA.402046

Breaking the limit of 3D fluorescence microscopy by dual-stage-processing network

Hao Zhang[1+], Yuxuan Zhao[1+], Chunyu Fang[1+], Guo Li[1], Meng Zhang[2], Yu-Hui Zhang[2,3], Peng Fei[1*]

[1]School of Optical and Electronic Information-Wuhan National Laboratory for Optoelectronics, Huazhong University of Science and Technology, Wuhan, 430074, China.

[2]Britton Chance center for Biomedical Photonics, Wuhan National Laboratory for Optoelectronics, Huazhong University of Science and Technology, Wuhan 430074, China.

[3]MoE Key Laboratory for Biomedical Photonics, Huazhong University of Science and Technology, Wuhan, 430074, China

[+] These authors contribute equally to this work

[*] feipeng@hust.edu.cn

Supplementary Information

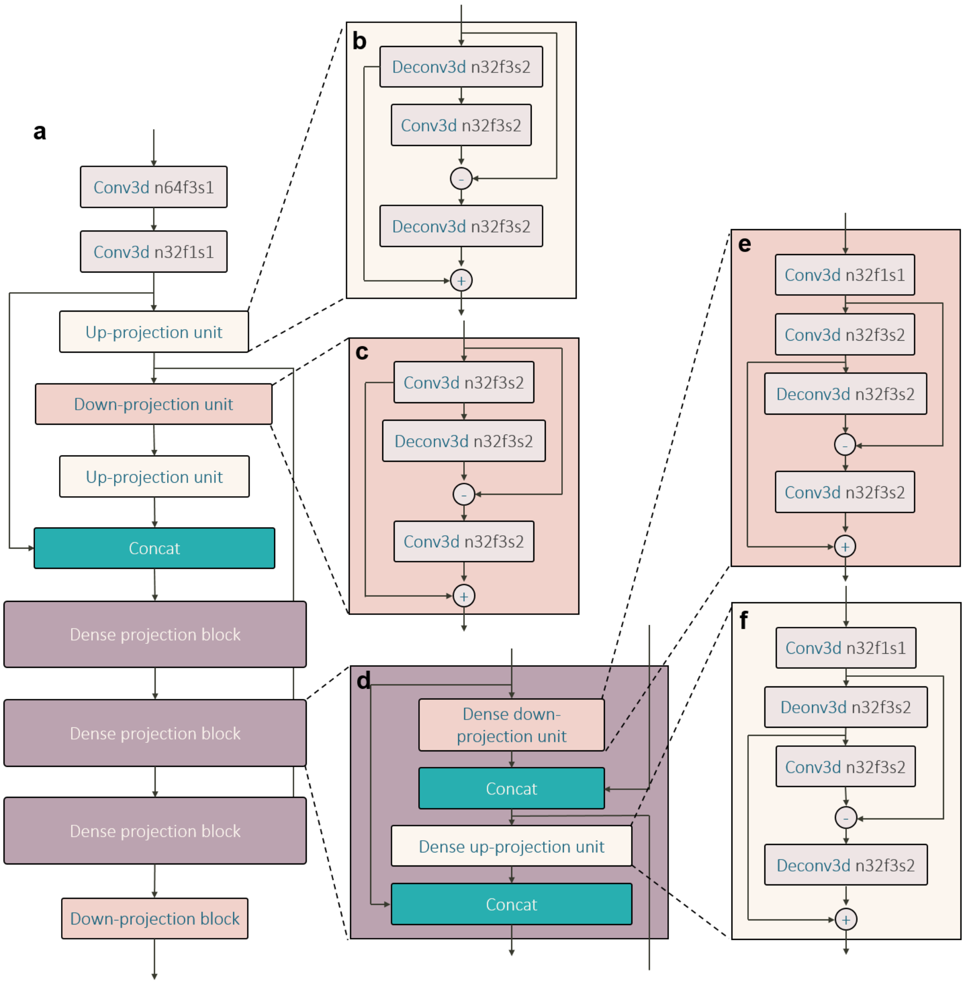


Fig. S1. **Deep back-projection network (DBPN) as the resolver of our DSP-Net. a,** The 3D DBPN architecture that contains several up-projection and down-projection units. Each convolutional layer in the units is characterized by the number of the filters n, the filter size f and the stride s. **b,** The structure of an up-projection unit that up-scales the size of input by a factor of 2 in each dimension. **c,** The structure of a down-projection unit that down-scales the size of input by a factor of 2 in each dimension. **d,** The structure of a dense projection block. It contains a dense down-projection unit, a dense up-projection unit and their concatenations. The "-" and "+" in the circle denote an element-wise subtraction and addition operation, respectively. **e,** The structure of a dense down-projection unit with an additional convolutional layer added, as compared to the regular down-projection unit. **f,** The structure of a dense up-projection unit with an additional convolutional layer added, as compared to the regular up-projection unit. Each convolutional layer uses the rectified linear unit (ReLU) as the active function.

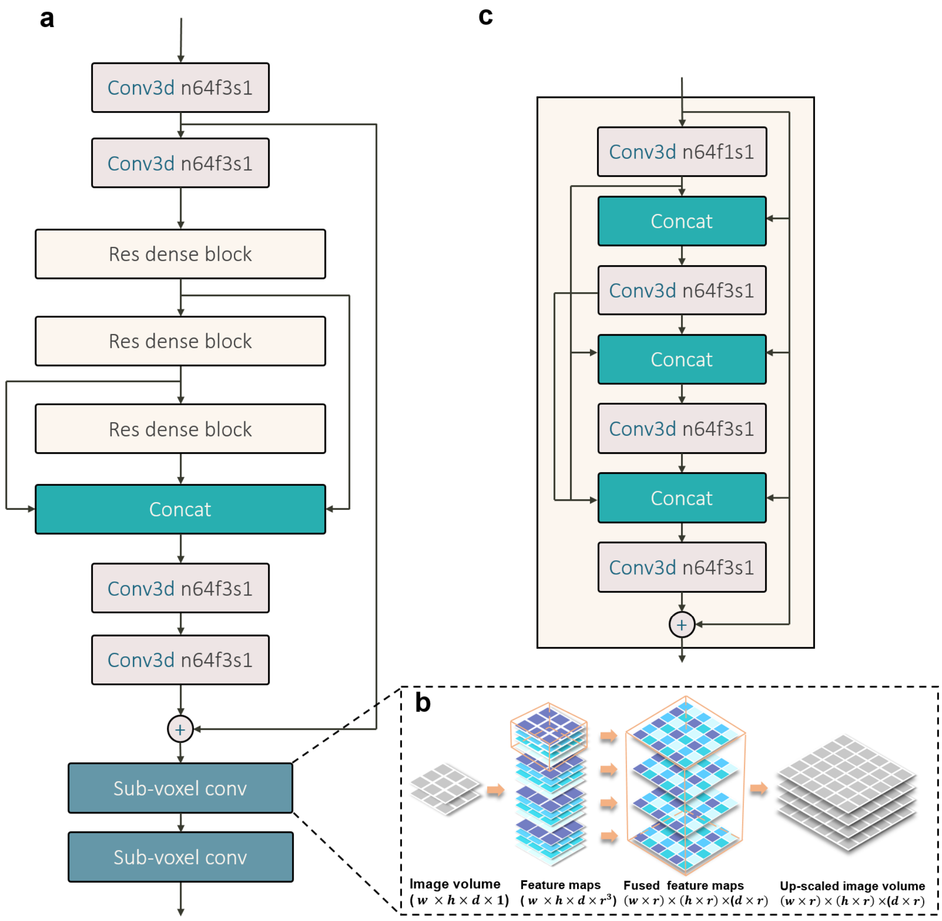


Fig. S2. Residual dense network (RDN) as the interpolator of our DSP-Net. a, The 3D RDN architecture that contains several residual dense blocks and sub-voxel convolutional layers. **b,** Details of a sub-voxel convolutional layer. Each sub-voxel convolutional layer interpolates the input image by a factor of r in all 3 dimensions. The following convolution layer with r^3 channels generates r^3 feature maps based on variant kernels. These feature maps are then integrated into a single channel, with adjacent channels merged into one to increase the size of the height, width and depth. Assuming that the single-channel input has a size of $w \times h \times d$ voxels, the final output has a size of $(w \times r) * (h \times r) \times (d \times r)$ voxels. **c,** The structure of a residual dense block in the RDN. The output of each convolutional layer is activated by ReLU.

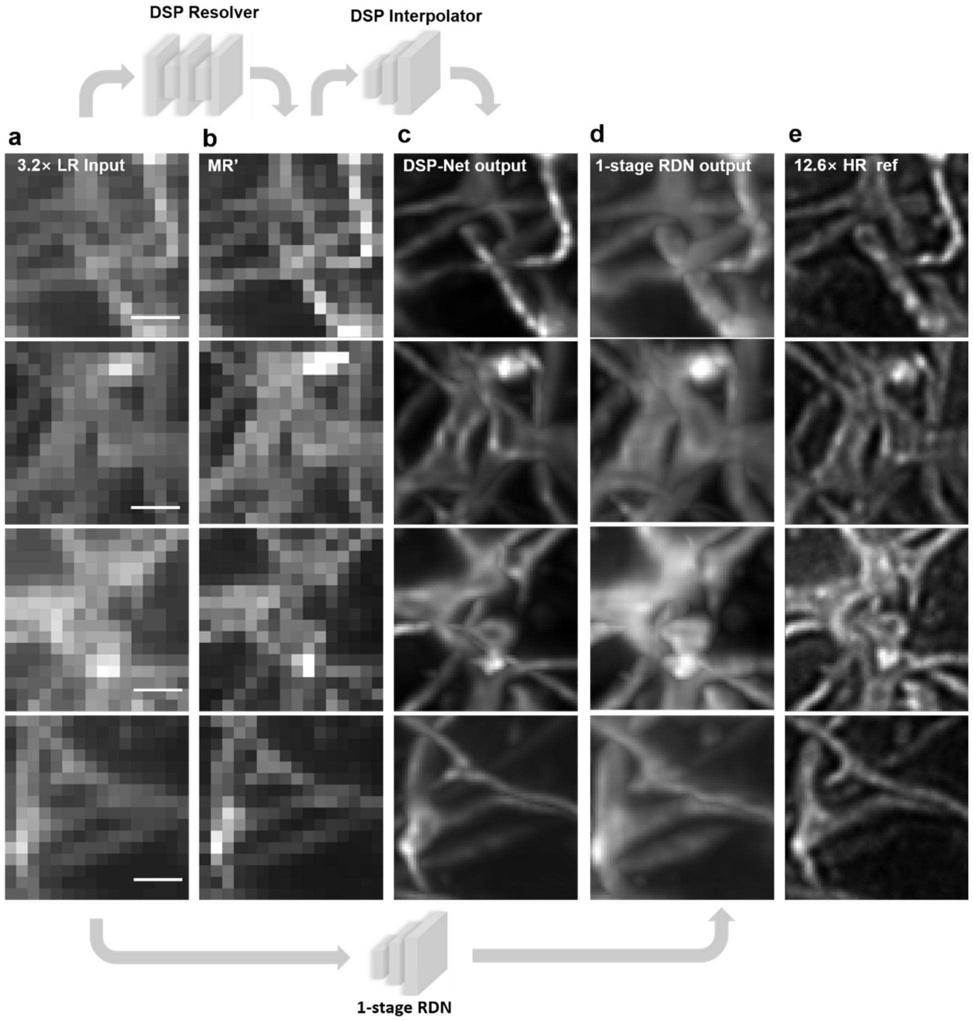


Fig. S3. DSP-Net recovering high-resolution details and suppressing background noises for the LSFM image of mouse brain vessels. The addition of the resolver together with the presence of intermediate result MR' in our DSP-Net can substantially enhance the potential signals while suppressing the background noises for the 3.2× LR inputs. As a result, the DSP-Net can finally restore more vessel structural details from the same raw LR inputs, as compared to the one-stage RDN which directly infers the outputs from the LR inputs. Both the RDN and DSP-Net results are compared with the HR references (12.6× images). **a**, The 3.2× inputs. **b**, The MR's by the resolver of the DSP-Net. **c**, The RDN outputs. **d**, The outputs of the standard one-stage RDN network. **e**, The 12.6× as the ground truths. Scale bars are 10 μm.

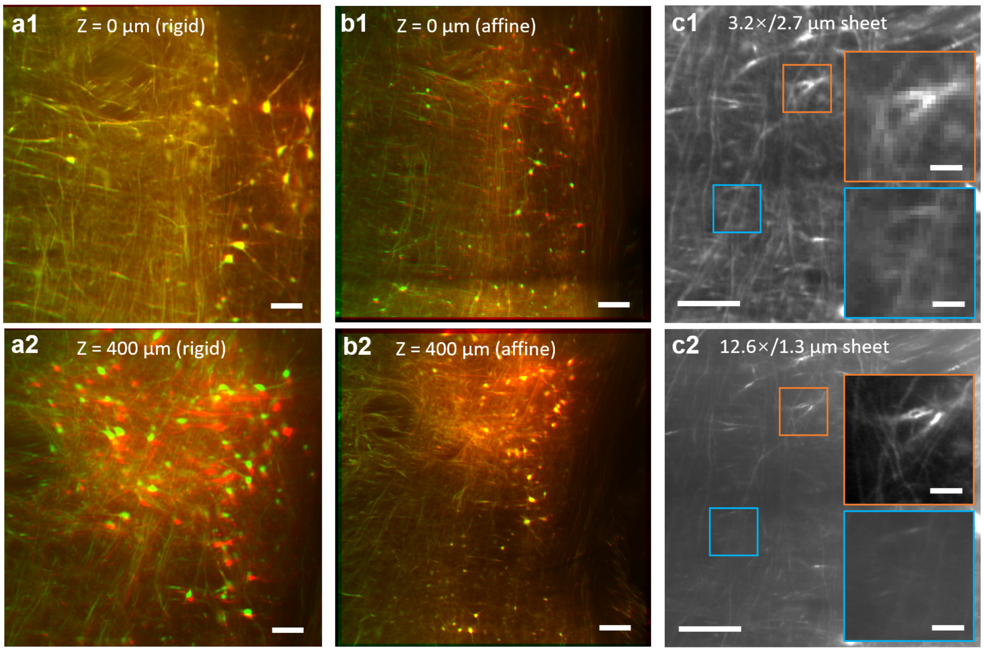


Fig. S4. Misalignments in 3D registration of LR and HR 3D images of mouse brain neurons experimentally obtained by 3.2× and 12.6× Bessel sheet modes. We registered the 3.2× LR and 12.6× HR 3D image stacks (500 × 500 × 400 μm volume) of cerebellum using Fiji registration plugin. The results shown in **a** and **b**, however, are far below the criteria of voxel-wise alignment, which is required by the neural network training. **a** shows the overlapping of the 3.2× (red channel) and 12.6× (green channel) image stack after registration using rigid model. While the neuronal signals at the most superficial layer (z = 0 μm, **a1**) of the registered volume are well aligned, the signals at the deepest layer (z = 400 μm) show obvious misalignment. This issue also exists in the 3D registration result by affine mode (**b**). It's known that the varying aberrations under different magnification factors could contribute to the difficulty for accurate image registration. In addition, the significant resolution gap in three dimensions also brings extra difficulty for highly-accurate alignment. As shown in **c**, the 3.2× LR slice obtained by thick 2.7-μm light-sheet excitation contains excessive signals (blue and orange boxes in **c1**) which are absent in the HR slice obtained by thinner 1.3-μm light-sheet excitation (blue boxes in **c2**). Scale bars are 50 μm in **a** to **c**, and 10 μm in insets. In our study, considering the abovementioned pitfalls for highly-accurate registration of measured HR and LR 3D images, we thus used half-synthetic data generated by our degradation model for the DSP-Net training.

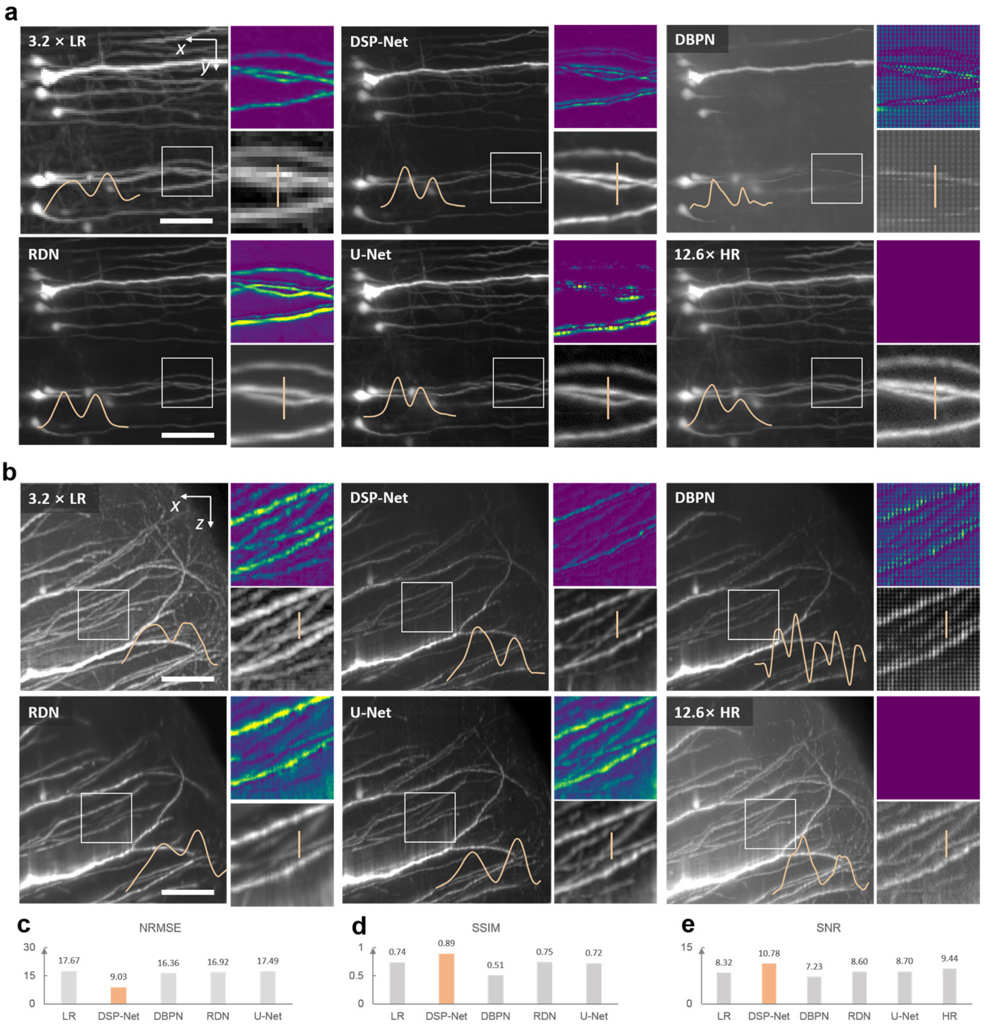


Fig. S5. Comparison between the DSP-Net and other state-of-the-art networks for high-throughput LSFM imaging of mouse brain. We applied DSP-Net to the recovery of a mouse cortex region originally imaged with 3.2× Bessel sheet, and compared the results with those from current one-stage networks. All the networks were trained on the same dataset with the same hyper-parameters (e.g., learning rate, batch size and epochs). Considering the U-Net doesn't up-scale the image size, its resolution enhancement process was performed on the bicubic interpolation of the inputs. **a,** The MIPs in *xy* planes of the selected region (200 × 200 × 200 μm) by 3.2× Bessel sheet + deconvolution, 3.2× Bessel sheet + DSP-Net, 3.2× Bessel sheet + DBPN, 3.2× Bessel sheet + RDN, 3.2× Bessel sheet + U-Net, and 12.6× Bessel sheet. **b,** The MIPs in *xz* planes of the selected region (200 × 200 × 200 μm) by 3.2× Bessel sheet + deconvolution, 3.2× Bessel sheet + DSP-Net, 3.2× Bessel sheet + DBPN, 3.2× Bessel sheet + RDN, 3.2× Bessel sheet + U-Net, and 12.6× Bessel sheet. Two small region-of-interests (ROIs)

in *xy* and *xz* planes were selected to calculate the pixel-wise difference between the reconstruction results by each method and the 12.6× HR references (insets). The 3.2× LR images were first interpolated into the same size as the HR references, to calculate the pixel-wise difference. **c~d,** As compared to other state-of-the-art network methods, DSP-Net showed minimum reconstruction errors, as indicated by the lowest normalized mean square error (NMSE) and highest structural similarity (SSIM) values. **e,** DSP-Net resolved finest neuronal structures while achieved the highest signal-to-noise ratio (SNR). The number of fitting parameters, which also indicated the size of network, was 779K for DPBN, 2276K for RDN, 3055K for DSP-Net and 16482K for U-Net (3D version based on CARE implementation), respectively. Scale bar, 50 µm.

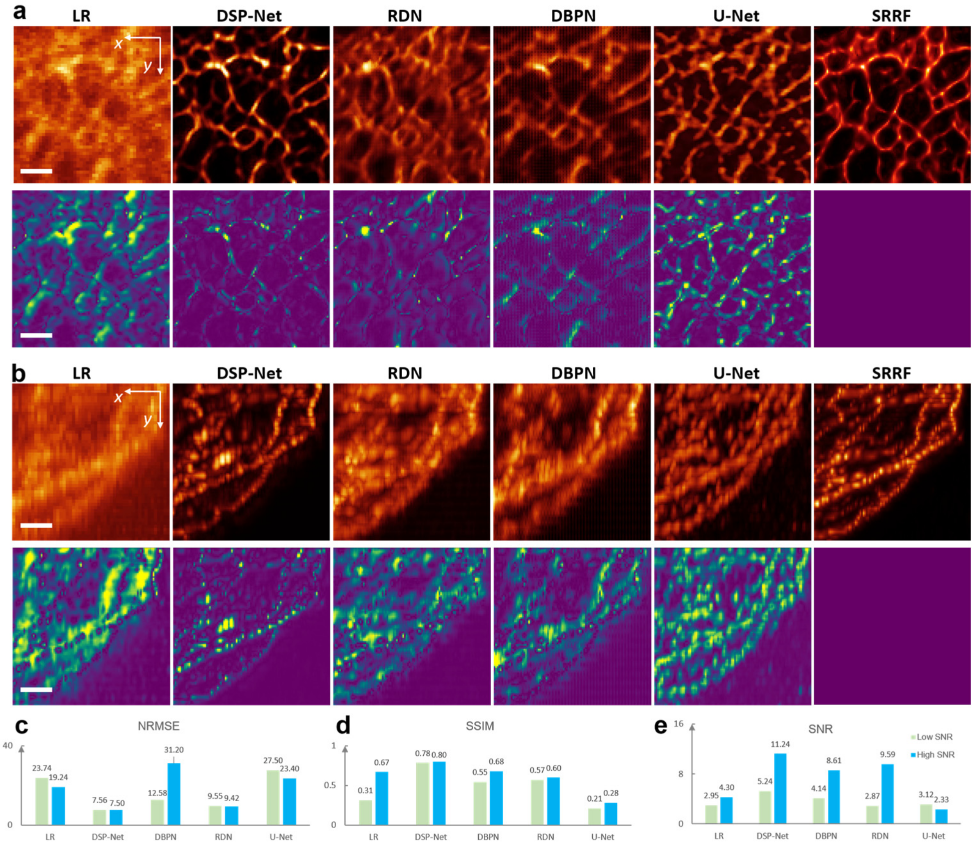


Fig. S6. Comparison between the DSP-Net and other state-of-the-art networks for imaging microtubules of *U2OS* cell beyond diffraction limit. a, We recovered a low-SNR diffraction-limited 3D image of *U2OS* cell (60× Bessel sheet) using DSP-Net, RDN, DBPN, U-Net, and finally compared the results with the 3-D SRRF result (*xy* planes in **a**, and *xz* planes in **b**). The error maps shown in the 2nd and 4th rows were generated by computing the pixel-wise difference between the outputs of each method and the 3-D SRRF results. The LR images were first interpolated into the same size as the HR references, to calculate the pixel-wise difference. As compared to other state-of-the-art network methods, the DSP-Net achieved highest-resolution, highest-fidelity (**c** and **d**) and highest-SNR (**e**) recovery for the diffraction-limited, low-SNR input. Scale bar, 2 µm.

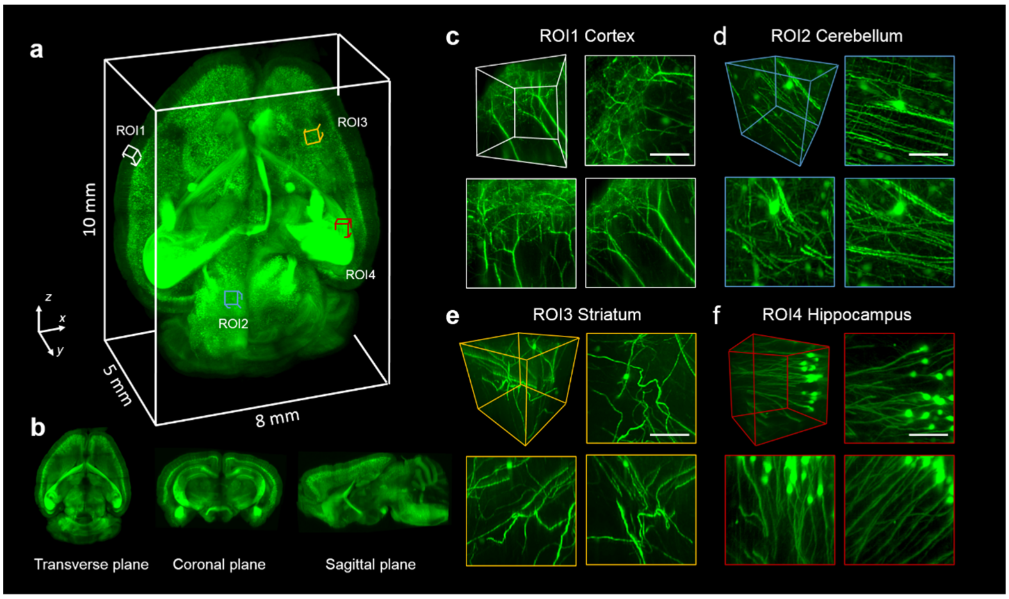


Fig. S7. DSP-Net reconstruction of neurons in whole mouse brain. The neuronal signals in a macro-scale whole brain show complicated distribution patterns as well as various intensities. Our DSP-Net can overcome these difficulties and restore high-quality signals at different areas of whole brain. **a,** The volume rendering of a DSP-Net reconstructed whole mouse brain with neurons labeled with GFP. **b**, The transverse (*xy*), coronal (*xz*) and sagittal (*yz*) planes of the brain, showing the complex signal distributions inside. **c-d,** Magnified views of four small ROIs in cortex (yellow), cerebellum (green), striatum (yellow) and hippocampus (red) regions, respectively. The raw brain images were rapidly acquired in merely ~6 minutes with using a 3.2× low-magnification objective plus a relatively thick 2.7-μm light-sheet illumination. The significant improvement by DSP-Net recovery allows the revealing of various neuron types/structures (dendrites of pyramidal neurons in **c**, astrocytes in **d**) at single-cell resolution (~1 μm).

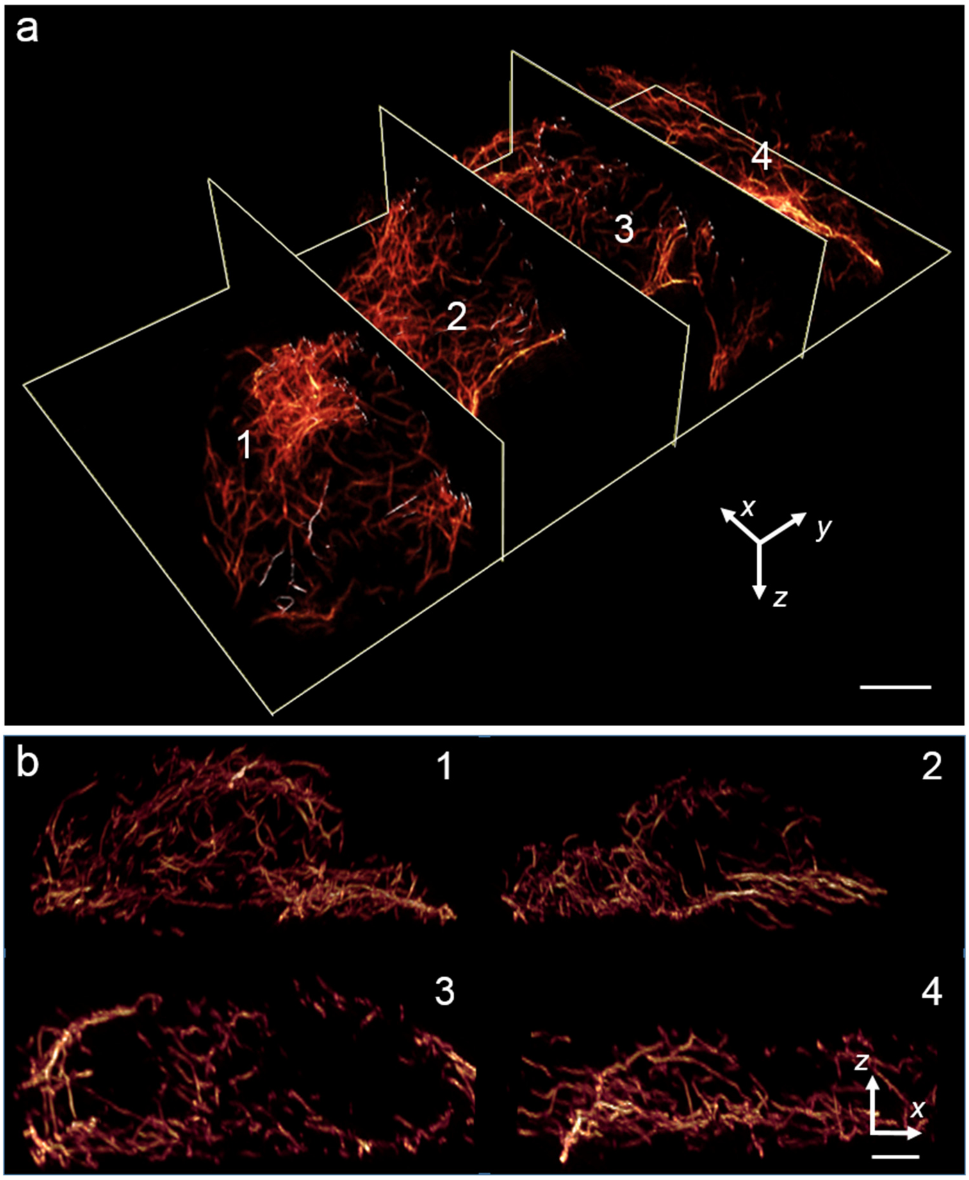


Fig. S8. DSP-Net reconstruction of single cell beyond diffraction limit. A *U2OS* cell was imaged by high-magnification Bessel light-sheet microscopy and super-resolved by our DSP-Net. **a**, 3D rendering of the whole cell. **b**, Slices through the cell along the planes. Scale bar: 10 μm.

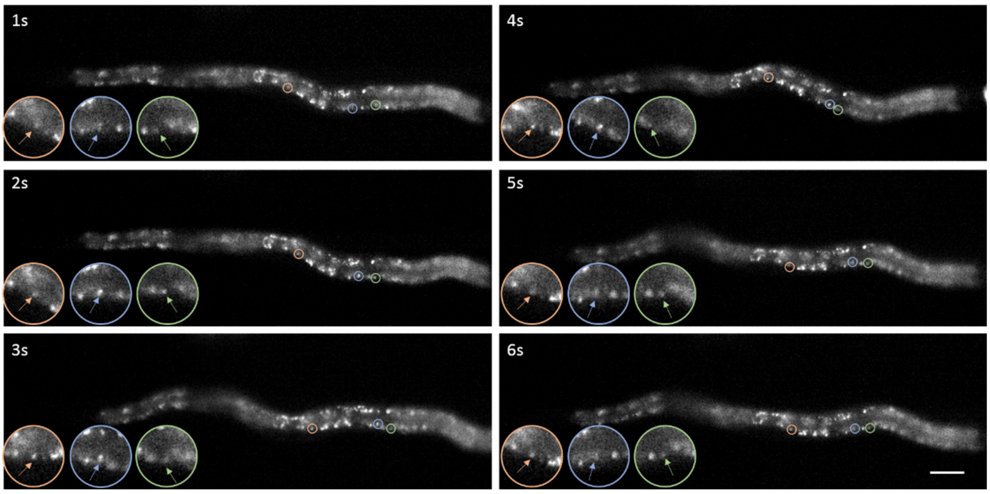


Fig. S9. Requirement of axial scanning in the *C. elegans* Imaging. We imaged the intestinal auto-fluorescence signals (with constant intensities) in freely moving *C. elegans* by 20× objective (Olympus, XLUMPLFLN20XW, 1.0 NA) with keeping the objective at a single focus plane. Due to the movement of the *C. elegans*, some signals defocus with obvious intensity change (indicated by the arrows), preventing the accurate tracking of the signal. Therefore, to record the 3D, time-varying *Ca^{2+}* signals accurately, it's necessary to rapidly scan the worm along z-direction. Scale bar: 40 μm.

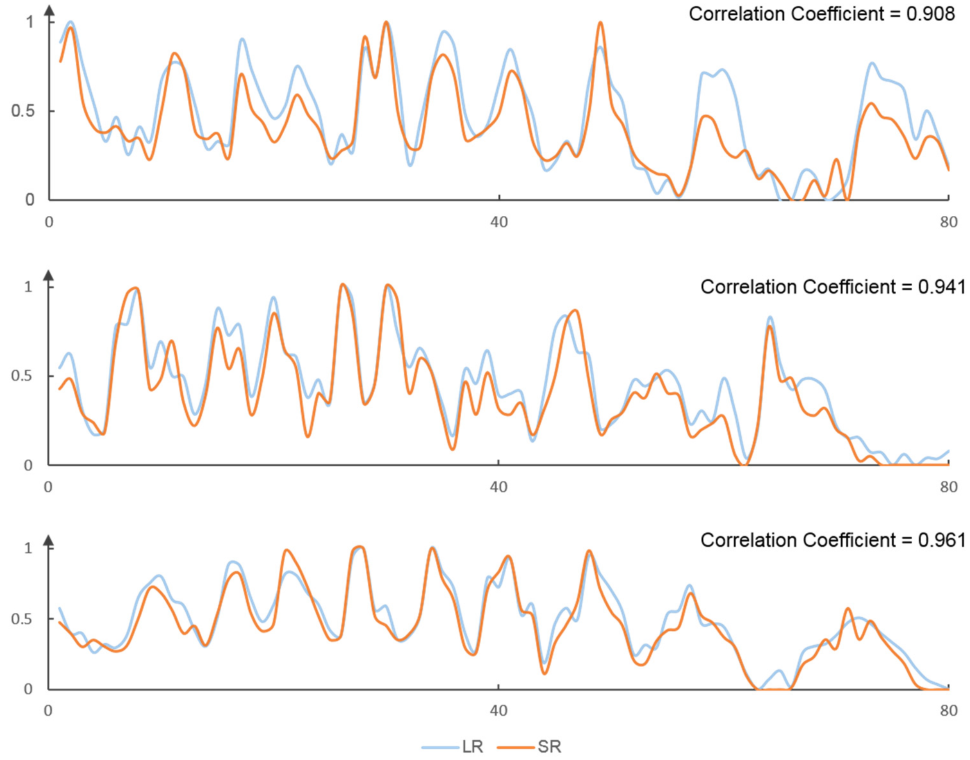


Fig. S10. Verifying the accuracy of the *Ca2+* signals reconstructed by DSP-Net. To verify that our DSP-Net provided quantitatively-accurate resolution-enhanced images, we generated LR 3D movie of a string of point-like signals with time-varying positions and intensities, to simulate the ground truth status of a GCaMP6-labelled acting worm. We then resolved the LR video using the DSP-Net, and analyzed the correlation coefficient between the time-varying *Ca2+* intensity of the LR signals and the DSP-Net reconstructed signals. The normalized intensities from three signal points in both raw video (blue) and SR video (red) were plotted against time. The DSP-Net results provided highly similar intensity fluctuation as compared to the raw signals, with showing correlation coefficients high than 90%. Therefore, we verified that a quantitatively accurate reconstruction of 3D signal intensities could be achieved by DSP-Net.

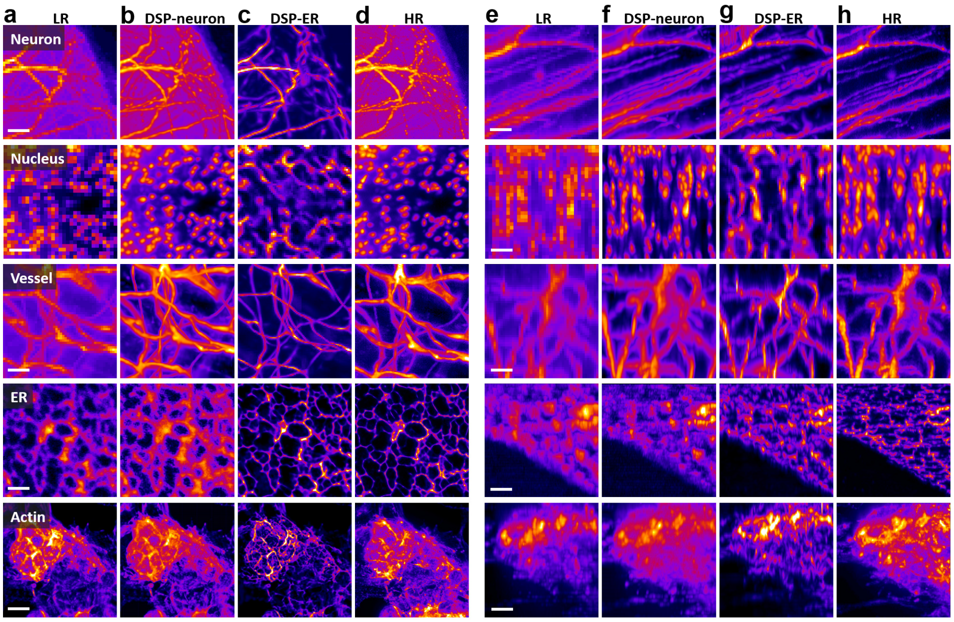


Fig. S11. Cross-sample cross-modality applications of the DSP-Net. We trained two DSP-Nets, termed DSP-neuron and DSP-ER, using data from the neurons of mouse brain (3.2× Bessel sheet as LR *versus* 12.6× Bessel sheet as HR) and endoplasmic reticulum (ER) of *U2OS* cell (diffraction-limited 60× Bessel sheet as LR *versus* 60× SRRF as HR), respectively. Then, we applied both networks to the resolution enhancement of a variety of biological samples, including the same types of mouse brain neurons and cell ER, as well as disparate types of mouse brain nuclei, mouse brain vessels imaged by 3.2× Bessel sheet, and actins of 3T3 cell imaged by 20× Bessel sheet. The MIP projections of the *xy* (left) and *yz* (right) planes from LR images, DSP-neuron reconstructions, DSP-ER reconstructions, and HR images are shown in **a** to **d**, and **e** to **h,** respectively. Both DSP-neuron and DSP-ER showed cross-sample cross-modality resolution enhancement capabilities for all the samples. However, the DSP-ER still performs better on the line-like structures of cells, such as ER (4th row) and actins (5th row), while the DSP-neuron shows higher-fidelity recovery for mouse nuclei (2nd row), neurons (1st row), and vessels (3rd row), in which the distributions of the signals were more diversified into point-like, line-like and belt-like shapes. These comparisons also indicated the limit of the cross-sample cross-modality capabilities of the network.

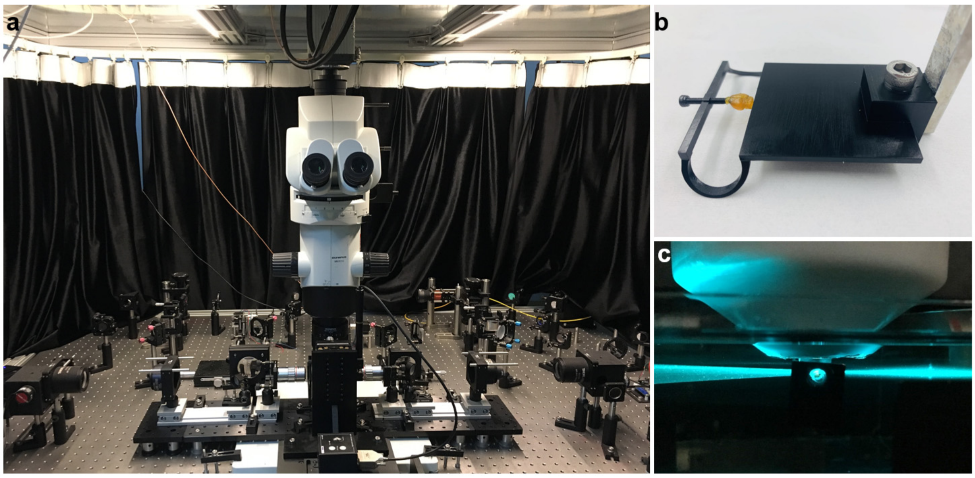


Fig. S12. Large-scale Bessel light-sheet setup for imaging neurons / vessels / nuclei in mouse brain. a, Photograph of our home-built large-scale Bessel light-sheet microscope with thin-and-wide Bessel sheet illumination (1.3 to 5 μm) and zoomable FOV (1.26 to 12.6×). **b**, Photograph of the customized sample holder that clamps the whole clarified mouse brain and dipped it into refractive index matching solution (RIMS). **c**, The work status of the Bessel sheet imaging of mouse brain. A scanned Bessel beam is forming a thin light-sheet to illuminate the brain. The rolling shutter of the camera was tightly synchronized with the scanning of Bessel beam, to eliminate the excitation from the side lobes.

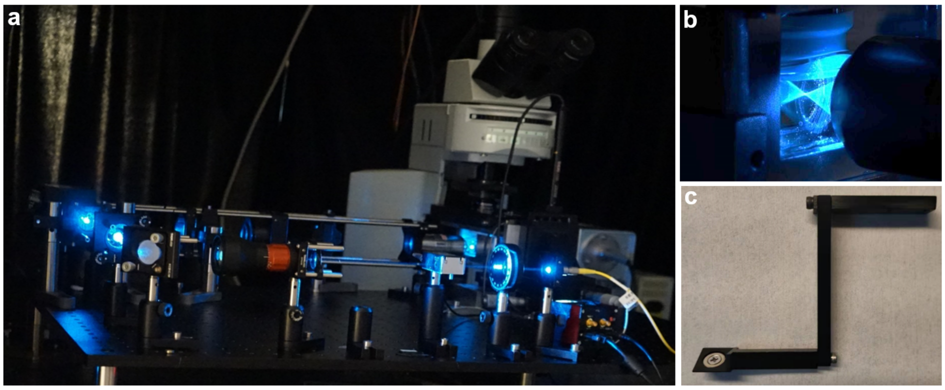


Fig. S13. High-magnification Bessel light-sheet setup for imaging ER / actins in *U2OS* and 3T3 cells. a, Photograph of our home-built high-magnification Bessel light-sheet microscope. **b**, The work status of the Bessel sheet imaging of single cell. **c**, The customized sample holder that clamps the glass slide with cell attached.

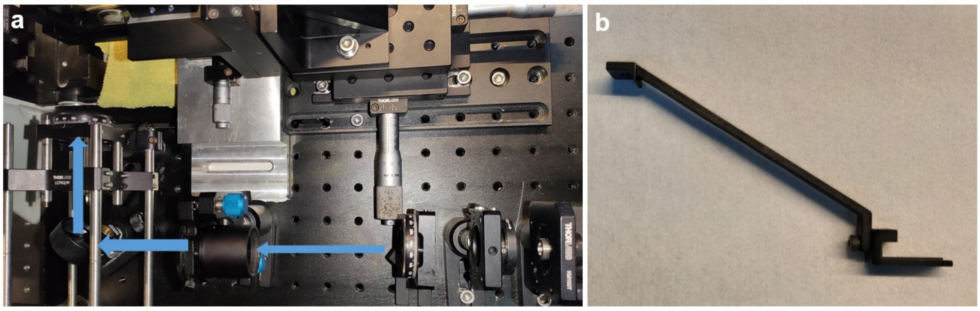


Fig. S14. High-speed Gaussian light-sheet setup for imaging freely-moving *C.elegans*. a, Photograph of plane illumination path, which generates a ~10 µm Gaussian light-sheet using a line-focusing cylindrical lens. **b**, The customized sample holder for mounting the microfluidic chip, inside which the worm can freely move.

Table S1. Parameters in imaging experiment and network training

Sample		Brain neurons		Brain vessels	Brain cell nuclei	Microtubules (*U2OS*)		Endoplasmic reticulum (*U2OS*)	*C.elegans*
Imaging mode		Bessel sheet	Confocal	Bessel sheet		Bessel sheet	Nikon-NSIM	Bessel sheet	Gaussian sheet
Magnification / NA	HR	12.6× / 0.5 1.3-μm sheet	12.6× / 0.5 (resampled from 16×/0.8)	12.6× / 0.5 1.3-μm sheet	8× / 0.38 1.3-μm sheet	SRRF	3D SIM	SRRF	Synthetic data
	LR	3.2× / 0.27 2.7-μm sheet	3.2× / 0.27 (degraded from 16×/0.8)	3.2× / 0.27 2.7-μm sheet	2× / 0.2 2.7-μm sheet	60×/1.1 0.8-μm sheet	100×/1.49	60×/1.1 0.8-μm sheet	4×/0.28 10-μm sheet
Voxel size (μm)	HR	0.5 × 0.5 × 1			0.8 × 0.8 × 1	0.055×0.055×0.15	0.03×0.03×0.1	0.055×0.055×0.15	0.41×0.41×1.1
	LR	2 × 2 × 4			3.25 × 3.25 × 4	0.11×0.11×0.3	0.06×0.06×0.2	0.11×0.11×0.3	1.63×1.63×4.5
Optical resolution (μm)	HR	0.62 × 0.62 × 2.7			1 × 1 × 2.7	/	0.105 × 0.105 × 0.13	/	/
	LR	1.15 × 1.15 × 1.3			1.89 × 1.89 × 1.3	0.29 × 0.29 × 0.8	0.21 × 0.21 × 0.27	0.29 × 0.29 × 0.8	1.11 × 1.11 × 10
Diffraction index		1.56 (BABB)	1.33 (water)	1.56 (BABB)		1.33 (PBS)	1.33 (PBS)	1.33 (PBS)	1.41 (56% glycerin)
Frame rate	HR	5 fps	1 fps	5 fps		0.167 fps	0.167 fps	0.167 fps	NA
	LR	20 fps		20 fps		5 fps	5 fps	5 fps	400 fps
Training data		Brain neurons	Brain vessels	Brain cell nuclei		Microtubules (*U2OS*)		Endoplasmic reticulum (*U2OS*)	*C.elegans*
Training patch size (voxels)	HR	80 × 80 × 80				96 × 96 × 32		64 × 64 × 24	96 × 96 × 40
	MR & LR	20 × 20 × 20				48 × 48 × 16		32 × 32 × 12	24 × 24 × 10
Training pairs		1137				696		424	112
Training time		13.75 h (500 epochs)				4.99 h (100 epochs)		7.01 h (200 epochs)	3.08 h (110 epochs)
Volume size		10 × 8 × 5 mm	0.63 × 0.63 × 0.3 mm	10 × 8 × 5 mm	5 × 8 × 5 mm	30 × 30 × 21 μm			3328 × 832 × 58 μm

Table S2. Throughput comparison in the imaging of mouse brains.

	3.2×	3.2×（DSP-Net）	6.4×	12.6×
Voxel size	2.03 μm×2.03 μm×4 μm	0.508 μm×0.508 μm×1 μm	1.015 μm×1.015 μm×1 μm	0.516 μm×0.516 μm×1 μm
Resolution	4.06 μm×4.06 μm×8 μm	1.016 μm×1.016 μm×2 μm	2.03 μm×2.03 μm×2 μm	1.032 μm×1.032 μm×2 μm
3D resolution	0.0078125 μm^{-3}	0.0625 μm^{-3}	0.5 μm^{-3}	0.5 μm^{-3}
Acquisition time	6 min (20 fps)	6 min	167 min (10 fps)	1200 min (5 fps)
Acquisition Throughput	83 megavoxels/s	5.4 gigavoxel/s	41.5 megavoxels/s	20 megavoxels/s
Imaging Speed	1.34×10^{9} μm^{3}/s	1.34×10^{9} μm^{3}/s	4.2×10^{7} μm^{3}/s	5.2×10^{6} μm^{3}/s
Stitching time	2 h	2 h	5 h	15 h
Data storage	150 Gb	3 Tb	500 Gb	3 Tb

Table S3. Throughput comparison between SRRF and DSP-Net in the imaging of *U2OS* cells.

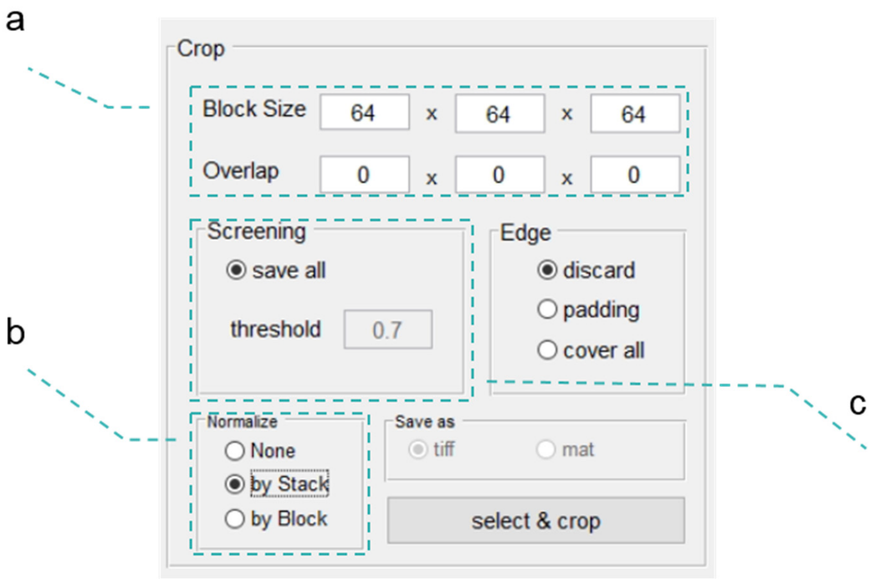
	SRRF	DSP-Net
Imaging time	19.40 s × 30	19.40 s
Reconstructing time	~ 480 s	24.57 s
Total time	~1062 s	43.97 s

Note S1. Implementation of DSP-Net.

This note elucidates the complete pipeline of DSP-Net processing. This deep learning based procedure can be roughly divided into three parts, as 1. Pre-preparation of training dataset, 2. Neural networks training, and 3. Resolution-enhancement inference.

1. Pre-preparation of training dataset

The training data for the DSP-Net are semi-synthetic, with either the LR inputs or the HR targets being generated computationally. In the former case, the HR 3D images are obtained via experiments, e.g., the 12.6×/0.5 Bessel sheet images of the mouse brain, while their LR pairs are generated using the abovementioned degradation model. For the single cell imaging, when the HR images beyond diffraction limit can't be acquired directly via experiment, we obtain them via computing a sequence of LR images (by 60×/1.1 Bessel sheet) using SRRF, and thereby pair the synthetic SRRF image with the LR measurements. As an intermediate part that bridges the significant resolution gap between the HR and the LR images, the MR images are usually down-sampled from the HR images. We use functions in MATLAB program to allow the easy operations of these image pre-preparation steps, where the image degradation is accomplished via applying the "imnoise" function followed by "imgaussfilt3" function, and the down-sampling is accomplished via applying "matrix re-slicing" function. Furthermore, considering the memory limits of the GPUs and the quality requirement for a high-accuracy mapping, the prepared 3D image pairs need to be further diced, normalized, and screened to generate small blocks suited for in-parallel network training on multiple GPUs. We also provide a program with GUI to readily execute these steps.



Block Size 64 x 64 x 64

Overlap 0 x 0 x 0

Screening: ● save all; threshold 0.7

Edge: ● discard ○ padding ○ cover all

Normalize: ○ None ● by Stack ○ by Block

Save as: ○ tiff ○ mat

Note Fig. S1. GUI of training dataset preprocessing program. The program integrates the following steps to generate appropriate 3D image blocks suited for DSP-Net training. **a**, Cropping a large-scale image stack into a number of small blocks. The size of each block and the overlap between the adjacent blocks (in voxels) are defined by user. **b**, Image normalization. The "None" option means the normalization is simply skipped. The "by Stack" option means the normalization is applied on the entire large-scale stack, while the "by Block" means that each subdivided block is normalized separately. **c**, Data screening. If the ratio button "save all" is unchecked, the program will delete the blocks where the maximum pixel values are smaller than the pre-set "threshold" value.

2. Network training

The DSP-Net uses the paired LR-HR image blocks for following iterative training. At each iteration, the DSP-Net takes one or several pairs of the training data, known as a "batch", to evaluate the loss functions and the corresponding gradients of the trainable parameters (i.e., the weights and the biases of the convolutional layers), and update these parameters along the minus direction of their estimated gradients. The training process goes through one "epoch" with finishing the calculation of all the batches, and repeats itself from the first batch for next new epoch. Usually, 20% images are reversed as the "testing data", for validating the performance of the network during training. When one epoch is finished, the network calculates the loss of the testing data, and saves the parameters of current model until the testing loss reaches its minimum. The training process lasts until either designated number of epochs are finished, or the model is not updated after a certain number of epochs. The DSP-Net finally converges to its best performance with the optimal model parameters recorded.

3. Raw data inference

The inference is based on a well-trained DSP-Net, and 3D low-resolution measurements of the samples. The network first reads the raw input images with automatically splitting them into small blocks according to the memory limit, and normalizing them into appropriate intensity scale. Then, with applying the optimized parameters obtained by previous training, the network deduces the resolution-enhanced output blocks and automatically stitches them back into a complete volume.

4. Image normalization

Before the images were fed into DSP-Net, they were normalized as following:

$$x = \frac{x}{max/2} - 1$$

Where *x* is the pixel intensity and *max* is the maximum intensity of the entire image (255 for 8-bit images and 65535 for 16-bit images). The pixel intensities of the images were then re-scaled into [-1, 1].

After being processed by the DSP-Net, the outputs were normalized by:

$$x = (x + 1) \times \frac{max}{2}$$

And the pixel intensities were re-scaled to [0, 255] or [0, 65535].

Note S2. Image degradation model.

The synthetic LR images for network training were artificially generated using a degradation model, which simulates the optical blurring of microscope and pixelization of camera. The implementation process can be described as following steps:

1. Gaussian blur:

According to the optics diffraction theory, an infinite small point would be blurred to a diffuse spot (Airy disk) known as the point spread function (PSF). Its radius of the first dark ring is given by $\sigma_{1-xy} = \frac{0.61\lambda}{NA_1}$. The lateral intensity distribution of the 3-D PSF in our optical systems conforms to the 1-order Bessel function and can be approximated to a Gaussian functions (Note Fig. S2). The axial intensity distribution of the Bessel beam conforms to the 0-order Bessel function, which can also be simulated by a Gaussian function. The radius of the first dark ring is a half of light-sheet thickness. Thus, the resolutions of the detection objective lenses used for HR and LR imaging can be represented by the radius of the corresponding Gaussian spots as:

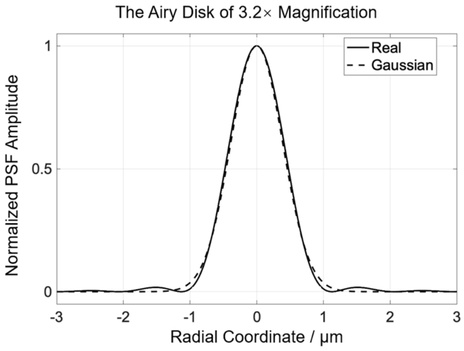
$$\sigma_{HR} = \frac{0.61\lambda}{NA_{HR}}$$

and

$$\sigma_{LR} = \frac{0.61\lambda}{NA_{LR}}$$

In the degradation model, the radius of the Gaussian function for blurring the HR can be deduced from:

$$\sigma_{degrade}^2 = \sigma_{HR}^2 - \sigma_{LR}^2$$



Note Fig. S2. Approximation of the Airy pattern using a Gaussian profile. A radial cross-section through the Airy pattern (solid curve) and its Gaussian approximation (dashed curve).

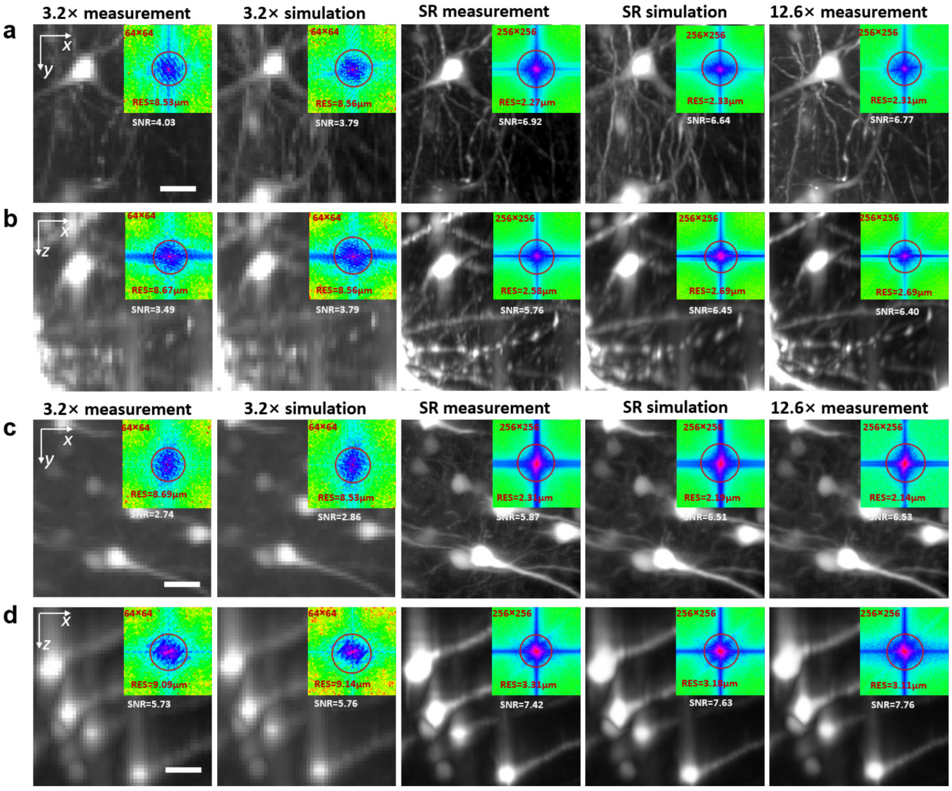
2. Down-sampling

The Gaussian blurring applied in the first step represents the optical blurring caused by the change of objective NA for LR and HR imaging. The second step of down-sampling operation accounts for the different degrees of pixelization by camera when sampling the optically-blurred object under different magnification factors, and by the different z step size when using different thickness of light-sheet illumination. As a reference, for generating the synthetic LR brain images using the HR images experimentally acquired by 12.6× objective plus 1-μm z-scan, we down-sampled the blurred HR image by 4 times in each dimension using a pixel average method, to simulate the LR measurement based on 3.2× objective plus 4-μm z step size.

3. Noise addition

At last to simulate the noise level of LR measurements, we add noises to the degraded images, including Gaussian and Poisson noise (simulating the CMOS noise) and perlin noise (simulating auto-fluorescence of samples). Since the Poisson uses the pixel intensity as the mean of its distribution, there are 4 parameter left to be determined: the mean value and the standard deviation of the Gaussian, and the octave and the persistence of the perlin. We use a grid search to traverse all the combinations of the parameters in pre-defined intervals until the synthetic LR images show very similar SNR and FFT spectrums to real LR measurements (Note Fig. S3).

To further verify the accuracy of our degradation model, we recovered both synthetic and measured LR images using the same trained DSP-Net and compared their SR outputs. As shown in the Note Fig. S3, the SR images deduced from synthetic and measured LR images show very high similarity.

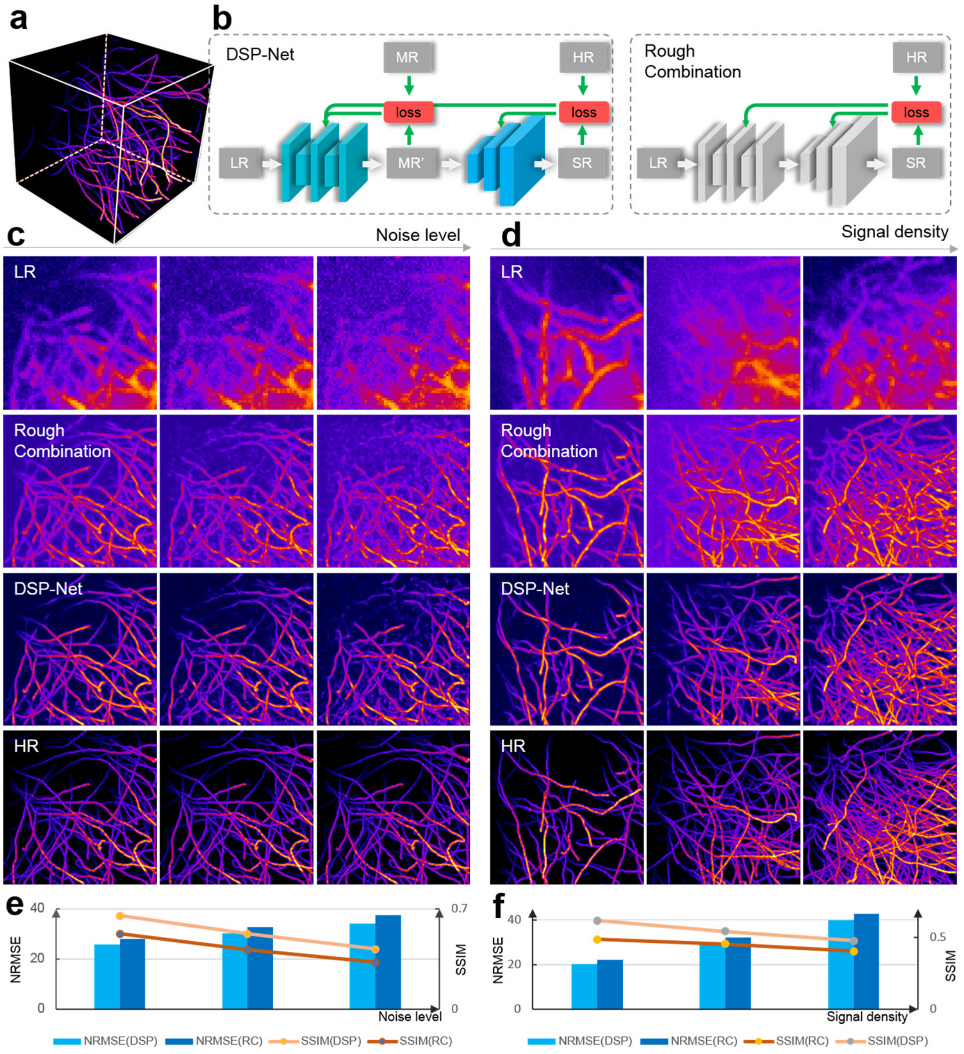


Note Fig. S2. Verification of the image degradation model. We visualized two 100 × 100 × 100 μm regions in cerebellum and cortex of a mouse brain, to verify the accuracy of our degradation model. We first compare the 3.2× LR simulations degraded from the 12.6× HR measurement, with the real 3.2× LR measurements. Then the SR results recovered from both the 3.2× LR simulations and 3.2× LR measurements are also compared. **a, b,** The comparison of MIPs in *xy* and *xz* planes of the cerebellum region. **c, d,** The comparison of MIPs in *xy* and *xz* planes of the cortex region. The insets in the projection images accordingly show their Fourier spectrums, from which the SNR and achieved resolution of the images are also calculated. The visually and quantitatively high similarity between both the low-resolution pairs (**3.2× measurement** and **3.2× simulation** results in blue box) and their corresponding SR reconstructions (**SR by measurement**, **SR by simulation,** and **12.6× HR measurement** in red box), verifies the sufficient accuracy of our image degrading model. Scale bar, 20 μm.

Note S3. Ablation study on dual-stage networks.

There remains a doubt that whether the image-enhancing capability of the DSP-Net comes from the introduction of MR and the resolving loss during training, or simply the result of the increased parameters through stacking two sub-nets. To address this concern, we conducted an ablation study by comparing the performance of the DSP-

Net and the rough combination (RC) of two sub-nets (without MR and the resolving loss when training), as shown in the following Note Fig. S4. We first generated HR, MR, and LR 3-D images of micro-tubulin. Then we trained the DSP-Net with HR-MR-LR datasets and the RC with HR-LR datasets, respectively. When both networks were converged, we used them to recover 1) LR inputs with different noise level, and 2) LR inputs with different signal density. The NRMSE and the SSIM for the DSP-Net outputs and for the RC outputs were calculated. Unsurprisingly, DSP-Net outperformed the RC on all testing conditions. DSP-Net showed not only perceptually clearer background, but also quantitatively better image quality (lower NRMSE, higher SSIM), which validated the significant advantage of our dual-stage design.



Note Fig. S3. Comparison between the DSP-Net and the rough combination (RC) of two sub-nets. **a,** The generated 3D micro-tubulin data. **b,** The training process of the DSP-Net and the

RC. There is not a resolving loss in RC to refine the 1st stage sub-net. **c,** MIPs of LR inputs of different noise level, the recovered results of DSP-Net and RC, and the corresponding HRs. **d,** MIPs of LR inputs of different signal density, the recovered results of DSP-Net and RC, and the corresponding HRs. **e**, The NRMSE and SSIM for the RC and for the DSP-Net outputs shown in **c**. **f,** The NRMSE and SSIM for the RC and for the DSP-Net outputs shown in **d**.