S.I Materials and methods (continued)

S.I.A Reflectance calibration and optical properties

SFDI reflectance images were obtained by phase-stepping a sinusoidal pattern onto the object plane, namely $S_i(x, y, f_x, \phi_i) = A_0 \cos(2\pi f_x x + \phi_i)$, with $\phi_0 = 0$, $\phi_1 = 2\pi/3$, and $\phi_2 = 4\pi/3$ radians. The modulation transfer function (MTF) of the tissue is captured by the AC component of illumination pattern at a specific optical wavelength and spatial frequency, commonly termed as $M_{AC}(x, y, f_x, \lambda)$, is obtained by combining the three phases at a single spatial frequency using a square-law detector

$$M_{AC}(x, y, f_x, \lambda) = \frac{\sqrt{2}}{3}\sqrt{(I_1 - I_2)^2 + (I_1 - I_3)^2 + (I_2 - I_3)^2}, \quad (1)$$

where $I_i = I(S_i(x, y, f_x, \phi_i), \lambda)$ is the optical intensity image captured when the projector is displaying $S_i(x, y, f_x, \phi_i)$ on the sample. This process is commonly referred to as *demodulation*. From these MTF images, the calibrated reflectance function $R_d(f_x, \lambda)$ of a sample can be obtained by eliminating the MTF of the imaging system with a reference material of known absorption and scattering properties, which was imaged under identical exposure and lighting conditions; this measurement is referred to as $M_{AC,ref}$. Then, the calibrated reflectance of a sample at a specific wavelength and spatial frequency, $R_d(x, y, f_x, \lambda)$ can be obtained via

$$R_d(x, y, f_x, \lambda) = R_{d,ref}(x, y, f_x, \lambda)\frac{M_{AC}(x, y, f_x, \lambda)}{M_{AC,ref}(x, y, f_x, \lambda)}, \quad (2)$$

and here $R_{d,ref}$ is a theoretical model of the reference material.

S.I.A.1 Diffusion approximation of the RTE

For a single pixel, the diffusion approximation of the RTE for air-turbid media interfaces is

$$R_d(f_x; \mu_s', \mu_a) = \frac{3Aa'}{\left(\frac{\mu_{eff}'}{\mu_{tr}} + 1\right)\left(\frac{\mu_{eff}'}{\mu_{tr}} + 3A\right)}, \quad (3)$$

where A is a proportionality constant that satisfies

$$A = \frac{1}{2}\left(\frac{1 - R_{eff}}{1 + R_{eff}}\right), \quad (4)$$

$$R_{eff} \approx 0.0636n + 0.668 + \frac{0.710}{n} - \frac{1.440}{n^2}, \quad (5)$$

$a' = \mu_s'/\mu_{tr}$ is the reduced albedo, $\mu_{tr} = \mu_a + \mu_s'$ is the transport coefficient, and $\mu_{eff}' = (\mu_{eff}^2 + k_x^2)^{1/2}$ with $\mu_{eff} = (3\mu_a\mu_{tr})^{1/2}$ is the modified effective transport coefficient, which is a function of both the spatial frequency of the projected pattern, f_x, and the optical wavelength, λ. Finally, n is the refractive index of the sample. Equation (3) assumes a semi-infinite air-turbid medium geometry with partial-current boundary conditions. Quantification requires finding a μ_s' and μ_a pair that provides a solution to the nonlinear least-squares fit problem that relates measured reflectance $R_d(f_x, \lambda)$ and Equation (3). Since this method is computationally expensive, additional procedures exist to increase curve fitting speed, namely LUT-based approaches [*S1*] and feedforward neural networks [*S2*, *S3*]. The latter option is optimal both in terms of precision and speed.

S.I.A.2 Sub-diffuse regime model

For spatial frequencies above $f_x \gg 0.33\mu_{tr}$ (i.e., ~ 0.5 mm^{-1} in breast tissue [*S4*]), the actual light transport specified by the RTE differs from the diffusion approximation by an order of magnitude, rendering Equation (3) inaccurate. Instead, a sub-diffuse light transport model is used, which assumes fewer scattering events and insensitivity to absorption. The detected signal is a function of the direction of individual backscattering events, revealing information about surface tissue microstructure [*S5*, *S6*]. This backscattered reflectance model, $R_{d,sd}$, also assumes a semi-infinite air-turbid medium and is given by

$$R_{d,sd}(f_x; \mu_s', \gamma) = \eta\left(1 + \left(\zeta_4\gamma^{-2}\right)v^{-\zeta_3\gamma}\right)\left(\frac{v^{\zeta_2\gamma}}{\zeta_1\gamma^2 + v^{\zeta_2\gamma}}\right), \quad (6)$$

where $v = \mu_s' f_x^{-1}$, and $\eta, \zeta_1, \ldots, \zeta_4$ are fitted parameters semi-empirically via Monte Carlo simulations of the medium. Specifically, $\zeta_1 = 68.6 \pm 3.3, \zeta_2 = 0.98 \pm 0.01, \zeta_3 = 0.61 \pm 0.01, \zeta_4 = 16.6 \pm 0.94$, and η is set to the insertion losses of the imaging system (for wide-field imaging, $\eta = 1$). Finally, the phase function parameter, $\gamma = (1 - g_2)/(1 - g_1)$, characterizes the relative probability of large backscattering events in the tissue, where g_1 and g_2 are the first and second Legendre moments of the scattering phase function $p(\theta)$ of the tissue under analysis [*S6*].

S.I.B Loss functions

The following loss functions are defined at different stages across the graph shown in Fig. 3(a)–(d) in the main document. We will refer to these expressions as *loss functions*, *cost functions* or *error functions* interchangeably throughout the article.

S.I.B.1 Primary autoencoder reconstruction loss

This network uses a combination of the $\mathcal{L}_2$ loss and an $\mathcal{L}_1$ *feature matching* loss [*S7*, *S8*] at the discriminator:

$$\mathcal{L}(r, \hat{r}) = \|r - \hat{r}\|^2 + \alpha\sum_{k=1}^{N_{\text{conv}}}|\psi_k(r) - \psi_k(\hat{r})|, \quad (7)$$

where $\psi_k(r)$ is all the feature maps at the k-th convolutional layer, for a total of N_{conv} layers in the discriminator, $D_{\text{aux}}(r)$. This term allows the network to learn textures eluded by the $\mathcal{L}_2$ distance. In all experiments, $\alpha = 0.05$, replicating previous work [*S9*].

S.I.B.2 Bottleneck constraints

Variational autoencoders apply an additional constraint to the bottleneck by comparing z's distribution $f_Z(z|x)$ with a known prior distribution $f_{Z,\text{target}}(z)$. For these experiments, two well-known expressions were tested: the conventional VAE loss [*S10*, *S11*]:

$$\mathcal{L}_{\text{VAE}}(z, f_{Z,\text{target}}(z)) = -\beta \cdot \mathcal{D}_{\text{KL}}\left(q(z|x)\|f_{Z,\text{target}}(z)\right), \quad (8)$$

with $\beta \in \mathbb{R}^+$ a constant for weighing the relevance of latent space Gaussianity versus reconstruction quality, $\mathcal{D}_{\text{KL}}(p\|q)$ the Kullback-Leibler divergence between distributions p and q, and the more recent Maximum Mean Discrepancy (MMD) loss [*S12*]:

$$\begin{aligned}\mathcal{L}_{\text{MMD}}(z, f_{Z,\text{target}}(z)) = &\mathbb{E}_{f_{Z,\text{target}}(z), f_{Z,\text{target}}(z')}\left[k(z, z')\right] \\ &+ \mathbb{E}_{f_Z(z), f_Z(z')}\left[k(z, z')\right] \\ &- 2\mathbb{E}_{f_Z(z), f_{Z,\text{target}}(z')}\left[k(z, z')\right]. \end{aligned} \quad (9)$$

In this expression, $k(.,.)$ is a Gaussian kernel function. Zhao *et al.*'s implementation for TensorFlow 1.14 was used in this manuscript (assuming $\alpha = 1.0$ in their decomposition of the ELBO objective, thereby prioritizing information preservation) [*S12*]. Autoencoders that use Equation 8 are usually referred to as β-Variational Autoencoders, or β-VAEs. Those using Equation 9 are defined as InfoVAEs or MMD-VAEs. The target distribution for both models is the standard normal Gaussian $f_{Z,\text{target}}(z) = \mathcal{N}(0, I)$.

S.I.B.3 Classifier

For classification, $\hat{y}$ will be $C(z)$'s prediction to datapoint z, and y will be its actual label. In that case, the conventional multiclass categorical cross-entropy loss will be used, as the output of the network will be a softmax layer representing predicted probabilities:

$$\mathcal{L}_{\text{cce}}(\hat{y}, y) = -\sum_{i=1}^{n} y_i \log(\hat{y}_i). \quad (10)$$

S.I.B.4 Nonlinear regressor

The OP estimation uses the Mean Squared Error on the OP vector $\mu \in \mathbb{R}^3$, namely $\mathcal{L}_{\text{MSE}}(\mu, \hat{\mu}) = \|\mu - \hat{\mu}\|^2$, where $\hat{\mu} = M(r_{\text{gen}})$.

S.I.B.5 Least-Squares GAN

Each individual GAN uses the Least-Squares GAN losses for the generator, $\mathcal{L}_G$, and the discriminators, $\mathcal{L}_{D_1}, \ldots, \mathcal{L}_{D_{N_{\text{disc}}}}$,, respectively [*S13*, *S14*]:

$$\mathcal{L}_{\text{G}}(z_{\text{gen}}) = \sum_{k=1}^{N_{\text{disc}}}(D_k(z_{\text{gen}}) - 1)^2, \quad (11)$$

$$\mathcal{L}_{D_k}(z_{\text{gen}}) = D_k(z_{\text{gen}})^2 + (D_k(z_{\text{real}}) - 1)^2. \quad (12)$$

S.I.C Other technical considerations

Additional technical decisions were made to ensure that the networks operated adequately. Consider the following:

S.I.C.1 Primary autoencoder

The primary autoencoder is a fully-skip-connected, convolutional MMD-VAE with an auxilliary discriminator. Additionally, the network includes small fully-connected layers to connect feature maps directly to the MLP elements of encoder and decoder, which we show speeds up learning. These layers substitute typical global averaging and/or flatten layers typical of standard architectures. An example is depicted in Fig. S1, with its tabular representation in Table 1.(a). Actual network dimensions are provided in Table 1.(b). Bottleneck size is $n_z = 256$. The decoder is comprised of consecutive, linearly-increasing xy-wise resizing of all previous layers, followed by a convolution operation, similarly to Gradually Upscaling Networks for image super-resolution [*S20*]. The model is trained with the reconstruction loss function described in Equation 7, and the bottleneck constraints from Equations 8 and 9. The rationale behind these choices is explained experimentally in Section III.A in the main manuscript.

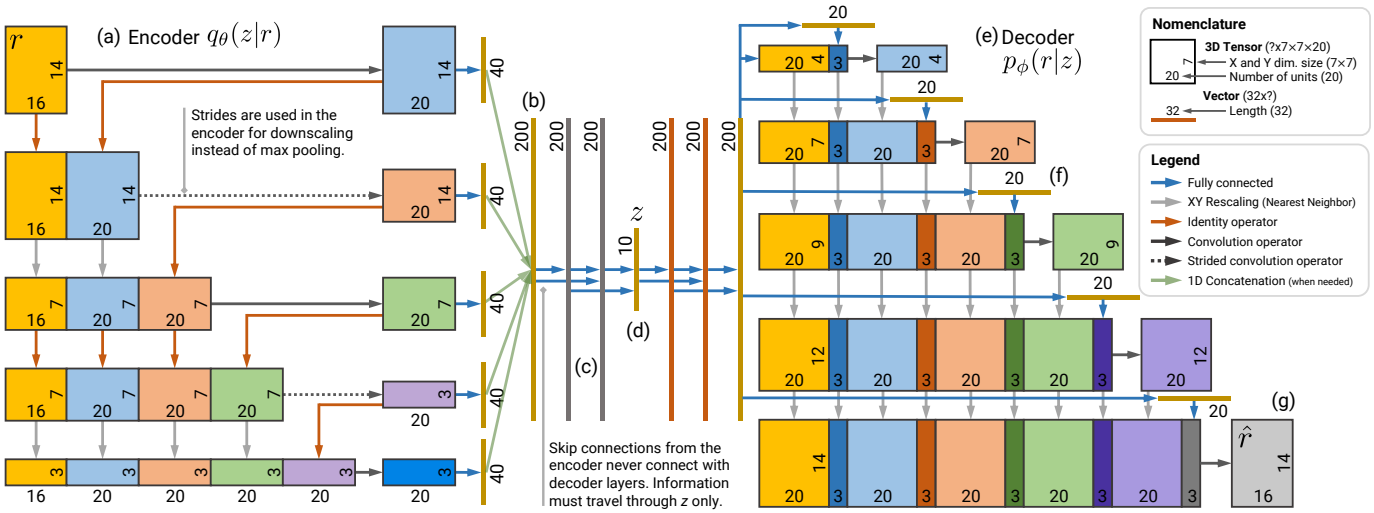

Figure S1: Example diagram of the final primary autoencoder architecture, as a result of combining current scientific literature for skip-connection multilayer perceptron VAEs, convolutional DenseNets, CoordConv layers, and all-convolutional nets [*S10*, *S11*, *S12*, *S15*, *S16*, *S17*, *S18*]. These figures represent a side view of the network, similar to schematics used to depict U-Nets [*S19*]. The skip-connected convolutional part of the encoder (a) is connected to a skip-connected MLP network (b) by concatenating fully connected projections of each individual generated feature map at maximum resolution. A two-layer MLP encoder (c) is then used to generate bottleneck layer z (d), followed by an MLP decoder and a convolutional, skip-connected Gradually Upsampling Network (GUN) [*S20*]. Small fully-connected feature maps (f) are included to the concatenated input stack of each convolutional layer, helping each new upscaling and convolutional layers in feature generation of the final patch (g) [*S17*]. The tabulated version of this network is provided in Table 1.(a).

Table 1: Network architectures

(a) Fig. S1, encoder (↓)*	**(b) Primary AE and Auxilliary Discriminator (↓)***	**(c) Secondary AE (↓)**	**(d) Classifier (↓)**
$? \times 14 \times 14 \times 16$ **Input**	$? \times 31 \times 31 \times 16$ **Input**	$? \times 256$ **Input**	$? \times 256$ **Input (or** $? \times 49$**)**
1×20 ELU, f.s. 3×3	3×120 ELU, f.s. 3×3	8×500 ELU, dense, s.c.	4×400 ELU, dense, s.c.
1×20 ELU, f.s. 3×3, stride=2	1×120 ELU, f.s. 3×3, stride=2	$? \times 2$ **Bottleneck, linear**	$? \times n_{\text{cls}}$ **Output, softmax**
1×20 ELU, f.s. 3×3	3×120 ELU, f.s. 3×3	8×500 ELU, dense, s.c.	**(e) OP estimator (↓)**
1×20 ELU, f.s. 3×3, stride=2	1×120 ELU, f.s. 3×3, stride=2	$? \times 256$ **Output, linear**	$? \times 16$ **Input**
1×20 ELU, f.s. 3×3	3×120 ELU, f.s. 3×3		5×300 ELU, dense, s.c.
Map to $? \times 40$ ELU per feature map	1×120 ELU, f.s. 3×3, stride=2		$? \times 3$ **Output, sigmoidal (**)**
Concatenation, 200 units total	3×120 ELU, f.s. 3×3	**(f) LS-GAN Generator (↓)**	**(g) LS-GAN Discriminator(↓)**
2×200 ELU, dense, s.c.	Map to $? \times 150$ ELU per feature map	$? \times 100$ **Input noise**, $\sim \mathcal{N}(0, 1)$	$? \times 256$ **Input**
$? \times 10$ **Bottleneck, linear**	Concatenation, 2250 units total	5×300 ELU, dense, s.c.	5×300 ELU, dense, s.c.
	3×1024 ELU, dense, s.c.	$? \times 256$ **Output, linear**	$? \times 1$ **Output, linear**
	$? \times 256$ **Bottleneck, linear**		

(*) Only the input half of the primary autoencoder is provided in this table. The decoder is symmetrically identical in terms of units per layer and activations. However, upscaling is linearly increased from 4×4 to the output width and height at every layer. Tensor upscaling uses `tf.image.resize_images`. Where the decoder of Fig. S1 uses 3 auxiliary input feature maps per new input convolutional layer, (b) uses 10 auxiliary feature maps.

(**) The output sigmoidal layers in the OP estimator are then renormalized to the allowed ranges specified in Section II.D of the manuscript. This ensures no negative OP values and detectable saturation beyond tolerable numbers.

Notation: f.s. stands for *filter size* (per unit). Layer notation is $c \times n$, $c :=$ number of identical consecutive layers, $n :=$ number of hidden neurons in each layer. Finally, *stride* indicates stride steps, if there are any. Input/output tensor shape notation as in TensorFlow: `(batch, height, width, channels)`. The token '?' denotes an unknown input batch size.

S.I.C.2 2D visualization autoencoder

The secondary autoencoder is a skip-connected, multi-layer perceptron MMD-VAE. Similarly to the primary autoencoder, all layers are connected to each other, with the exception of the bottleneck ($n_{z'} = 2$). No skip-connections are allowed to cross the bottleneck. This type of structure is known to allow the encoder and decoder layers to adapt to specific problems, improving reconstruction fidelity [*S18*]. The number of layers and units per layer is left in Table 1.(c). The model is trained to minimize $\frac{1}{2}\|z - \hat{z}\|^2$ and Equation 9.

S.I.C.3 Baseline autoencoder

To prove whether or not textural information improves separability among tissue categories, an additional autoencoder was prepared. Its architecture is identical to the secondary autoencoder of Table 1.(c), but with a 16-unit input (4 frequencies, 4 wavelengths). Intuitively, it is expected that tissue morphology (i.e. SFDI-enhanced textural contrast) will improve overall classification in more ambiguous cases where optical responses will be similar (this has been tested in practice with more traditional machine learning methods [*S21*]). To prove this notion in deep learning models, the aforementioned autoencoder will serve as a control baseline for separability without taking textural information into account.

S.I.C.4 Classifier

An example classifier is provided separately as a tentative study of classification accuracy, and it consists of a skip-connected MLP. Presented in Table 1.(d), it feeds from the primary autoencoder bottleneck z, and provides a classification based on a set of specified categories $\hat{y}$. The model is trained with a categorical cross-entropy loss (Equation 10).

S.I.C.5 Optical properties estimator

The estimator is another MLP with skip connections. This last network connects reflectance data with the optical properties model: $f_{\text{OP}} : r \rightarrow \hat{\mu}$, where $\mu = (\mu_s', \mu_a, \gamma)$. In these experiments we only evaluate pixel-wise OPs.

S.I.C.6 Skip connections

Generally, all the networks use skip connections, which are known to smooth out the loss function landscape [*S22*] and avoid Fisher information losses between layers [*S18*]. Skip connections between layers were tested in the primary autoencoder, and the same decision was replicated in the rest of the networks. As demonstrated in Section III.A, skip connections provide an improved degree of generalization and faster convergence than networks without them [*S18*].

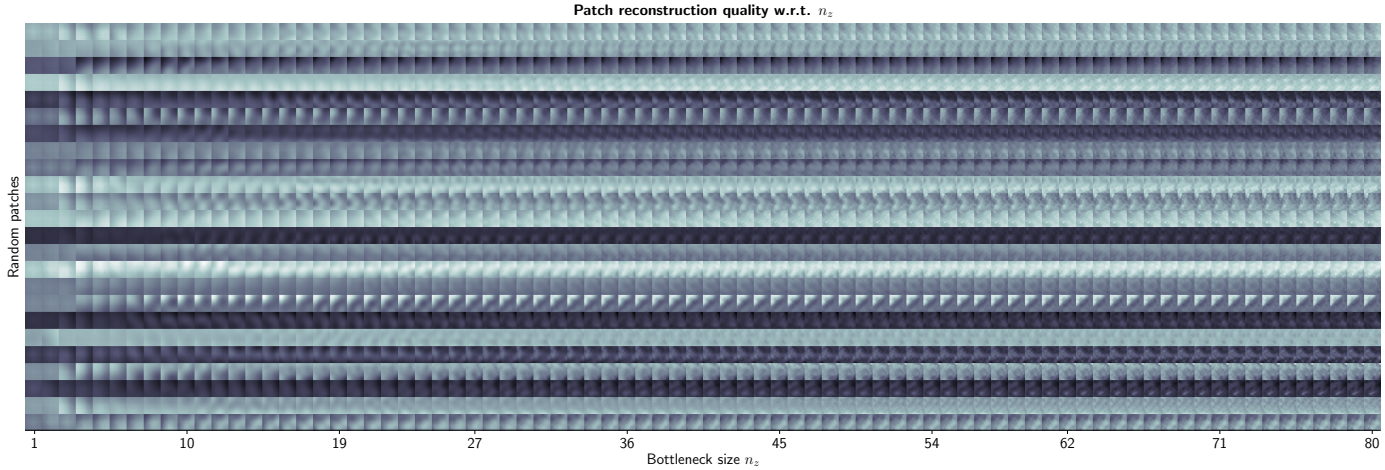

Figure S2: Close-up detail for patch reconstructions for different values of n_z. This plot is identical to Figure 7.(c) in the Main Document but with more patches and up to $n_z = 80$ dimensions. Patches are selected randomly and are not cherry-picked. Best viewed in digital format.

S.I.C.7 Additional fully-connected layers

The primary autoencoder connects the middle MLP section of the network to convolutional elements via fully-connected layers. These connections can be seen in Fig. S1 linking feature maps in the encoder to the first fully connected layer. They are also visible in the decoder, as small feature maps that are concatenated to the input of each additional convolutional layer.

S.I.C.8 Dropout regularization

Dropout is a well-known method for minimizing overfitting in neural networks [*S23*]. Generally, dropout is used with different drop values depending on the network and simulation, as specified in Table III in the main document.

S.I.C.9 Activation functions

The ELU activation function was employed in all hidden units, given its faster convergence and approximate batch normalization properties, while avoiding *ReLU death* and gradient vanishing across layers [*S24*]. ELU units are inhibited or signify the presence of some feature, with an improved resilience to noise. Classifier networks are provided with an output softmax layer, and all other output layers (such as the bottlenecks and outputs in the AEs, the optical estimator network, and LS-GANs) are linear units. The auxilliary discriminator $D_{\text{aux}}(r)$ uses a different activation function –leaky ReLUs—as a way to hinder its learning speed while still avoiding ReLU death (convergence was not achieved otherwise). The OP estimator also uses leaky ReLUs for a slower, more stable convergence.

S.I.C.10 Hardware used

A total of 5 machines were employed. The first three were Docker containers running in parallel, each with a 64-core Intel Xel Gold 6230 CPU and an nVidia Tesla v100 GPU (32 GB of VRAM), with up to 384 GB of shared RAM and sufficient storage, which were used for large-network training, namely Sections III.A, III.D and III.E in the main article. The other two were equivalent desktop computers, one with an Intel Core i9-9700K CPU and an nVidia RTX 2080Ti GPU, with 64 GB of RAM, and a second computer with an AMD Ryzen 5 3600 CPU, an nVidia 2080 Super GPU, and 32 GB of RAM. The latter were used on inference for all Sections and training the OP estimator, which required less compute.

References

[S1] D. J. Cuccia, F. Bevilacqua, A. J. Durkin, F. R. Ayers, and B. J. Tromberg, "Quantitation and mapping of tissue optical properties using modulated imaging," *J. Biomed. Opt.*, vol. 14, no. 2, p. 024012, 2009.

[S2] Y. Zhao, Y. Deng, F. Bao, H. Peterson, R. Istfan, and D. Roblyer, "Deep learning model for ultrafast multifrequency domain optical property extractions for spatial frequency domain imaging," *Opt. Lett.*, vol. 43, no. 22, pp. 5669–5672, 2018.

[S3] A. C. Stier, W. Goth, Y. Zhang, M. C. Fox, J. S. Reichenberg, F. C. Lopes *et al.*, "A machine learning approach to determining sub-diffuse optical properties," in *Biophotonics Congress: Biomedical Optics 2020 (Translational, Microscopy, OCT, OTS, BRAIN)*. Optical Society of America, 2020, p. SM2D.6. [Online]. Available: http://www.osapublishing.org/abstract.cfm?URI=OTS-2020-SM2D.6

[S4] D. M. McClatchy-III, E. J. Rizzo, W. A. Wells, P. P. Cheney, J. C. Hwang, K. D. Paulsen *et al.*, "Wide-field quantitative imaging of tissue microstructure using sub-diffuse spatial frequency domain imaging," *Optica*, vol. 3, no. 6, pp. 613–621, 2016.

[S5] V. Krishnaswamy, J. T. Elliot, D. M. McClatchy, R. J. Barth, W. A. Wells, B. W. Pogue *et al.*, "Structured light scatteroscopy," *J. Biomed. Opt.*, vol. 19, no. 7, 2014.

[S6] S. C. Kanick, D. M. III, V. Krishnaswamy, J. T. Elliott, K. D. Paulsen, and B. W. Pogue, "Sub-diffusive scattering parameter maps recovered using wide-field high-frequency structured light imaging," *Biomed. Opt. Express*, vol. 5, no. 10, pp. 3376–3390, 2014.

[S7] M. E. A. Seddik, M. Tamaazousti, and J. Lin, "Generative collaborative networks for single image super-resolution," *Neurocomputing*, 10 2019.

[S8] Y. Mroueh, T. Sercu, and V. Goel, "Mcgan: Mean and covariance feature matching gan," *ArXiv*, vol. abs/1702.08398, 2017.

[S9] M. Mathieu, C. Couprie, and Y. LeCun, "Deep multi-scale video prediction beyond mean square error," *CoRR*, vol. abs/1511.05440, 2016.

[S10] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *CoRR*, vol. abs/1312.6114, 2014.

[S11] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," in *Proceedings of the 31st International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, E. P. Xing and T. Jebara, Eds., vol. 32, no. 2. Bejing, China: PMLR, 22–24 Jun 2014, pp. 1278–1286. [Online]. Available: http://proceedings.mlr.press/v32/rezende14.html

[S12] S. Zhao, J. Song, and S. Ermon, "InfoVAE: Information Maximizing Variational Autoencoders," *arXiv e-prints*, p. arXiv:1706.02262, Jun 2017.

[S13] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, "Least squares generative adversarial networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2813–2821.

[S14] I. Durugkar, I. Gemp, and S. Mahadevan, "Generative multi-adversarial networks," *ArXiv*, vol. abs/1611.01673, 2017.

[S15] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2261–2269.

[S16] G. Huang, S. Liu, L. van der Maaten, and K. Weinberger, "Condensenet: An efficient densenet using learned group convolutions," 06 2018, pp. 2752–2761.

[S17] R. Liu, J. Lehman, P. Molino, F. P. Such, E. Frank, A. Sergeev *et al.*, "An intriguing failing of convolutional neural networks and the coordconv solution," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, ser. NIPSâĂŹ18. Red Hook, NY, USA: Curran Associates Inc., 2018, p. 9628âĂŞ9639.

[S18] H. Zheng, J. Yao, Y. Zhang, and I. W.-H. Tsang, "Degeneration in vae: in the light of fisher information loss," *ArXiv*, vol. abs/1802.06677, 2018.

[S19] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds. Cham: Springer International Publishing, 2015, pp. 234–241.

[S20] Y. Zhao, G. Li, W. Xie, W. Jia, H. Min, and X. Liu, "Gun: Gradual upsampling network for single image super-resolution," *IEEE Access*, vol. 6, pp. 39 363–39 374, 2018.

[S21] S. S. Streeter, B. W. Maloney, D. M. McClatchy, M. Jermyn, B. W. Pogue, E. J. Rizzo *et al.*, "Structured light imaging for breast-conserving surgery, part II: texture analysis and classification," *Journal of Biomedical Optics*, vol. 24, no. 9, pp. 1 – 12, 2019. [Online]. Available: https://doi.org/10.1117/1.JBO.24.9.096003

[S22] H. Li, Z. Xu, G. Taylor, and T. Goldstein, "Visualizing the loss landscape of neural nets," in *NeurIPS*, 2017.

[S23] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *ArXiv*, vol. abs/1207.0580, 2012.

[S24] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," *arXiv preprint arXiv:1511.07289*, 2015.