Advances in Electron Microscopy with Deep Learning

by

Jeffrey Mark Ede

Thesis

Submitted to the University of Warwick

for the degree of

Doctor of Philosophy in Physics

Department of Physics

February 2021

Contents

List of Abbreviations

AE	Autoencoder
AFM	Atomic Force Microscopy
ALRC	Adaptive Learning Rate Clipping
ANN	Artificial Neural Network
ASPP	Atrous Spatial Pyramid Pooling
A-tSNE	Approximate t-Distributed Stochastic Neighbour Embedding
AutoML	Automatic Machine Learning

Bagged	Bootstrap Aggregated
bfloat16	16 Bit Brain Floating Point
BM3D	Block-Matching and 3D Filtering
BPTT	Backpropagation Through Time

CAE	Contractive Autoencoder
CBED	Convergent Beam Electron Diffraction
CBOW	Continuous Bag-of-Words
CCD	Charge-Coupled Device
cf.	*Confer*
Ch.	Chapter
CIF	Crystallography Information File
CLRC	Constant Learning Rate Clipping
CNN	Convolutional Neural Network
COD	Crystallography Open Database
COVID-19	Coronavirus Disease 2019
CPU	Central Processing Unit
CReLU	Concatenated Rectified Linear Unit
CTEM	Conventional Transmission Electron Microscopy
CTF	Contrast Transfer Function
CTRNN	Continuous Time Recurrent Neural Network

CUDA	Compute Unified Device Architecture
cuDNN	Compute Unified Device Architecture Deep Neural Network
DAE	Denoising Autoencoder
DALRC	Doubly Adaptive Learning Rate Clipping
DDPG	Deep Deterministic Policy Gradients
D-LACBED	Digital Large Angle Convergent Beam Electron Diffraction
DLF	Deep Learning Framework
DLSS	Deep Learning Supersampling
DNN	Deep Neural Network
DQE	Detective Quantum Efficiency
DSM	Doctoral Skills Module
EBSD	Electron Backscatter Diffraction
EDX	Energy Dispersive X-Ray
EE	Early Exaggeration
EELS	Electron Energy Loss Spectroscopy
e.g.	*Exempli Gratia*
ELM	Extreme Learning Machine
ELU	Exponential Linear Unit
EM	Electron Microscopy
EMDataBank	Electron Microscopy Data Bank
EMPIAR	Electron Microscopy Public Image Archive
EPSRC	Engineering and Physical Sciences Research Council
Eqn.	Equation
ESN	Echo-State Network
ETDB-Caltech	Caltech Electron Tomography Database
EWR	Exit Wavefunction Reconstruction
FIB-SEM	Focused Ion Beam Scanning Electron Microscopy
Fig.	Figure
FFT	Fast Fourier Transform
FNN	Feedforward Neural Network
FPGA	Field Programmable Gate Array
FT	Fourier Transform
FT^{-1}	Inverse Fourier Transform
FTIR	Fourier Transformed Infrared
FTSR	Focal and Tilt Series Reconstruction

GAN	Generative Adversarial Network
GMS	Gatan Microscopy Suite
GPU	Graphical Processing Unit
GRU	Gated Recurrent Unit
GUI	Graphical User Interface
HPC	High Performance Computing
ICSD	Inorganic Crystal Structure Database
i.e.	*Id Est*
i.i.d.	Independent and Identically Distributed
IndRNN	Independently Recurrent Neural Network
JSON	Javascript Object Notation
KDE	Kernel Density Estimated
KL	Kullback-Leibler
LR	Learning Rate
LSTM	Long Short-Term Memory
LSUV	Layer-Sequential Unit-Variance
MAE	Mean Absolute Error
MDP	Markov Decision Process
MGU	Minimal Gated Unit
MLP	Multilayer Perceptron
MPAGS	Midlands Physics Alliance Graduate School
MTRNN	Multiple Timescale Recurrent Neural Network
MSE	Mean Squared Error
N.B.	*Nota Bene*
NiN	Network-in-Network
NIST	National Institute of Standards and Technology
NMR	Nuclear Magnetic Resonance
MSE	Neural Network Exchange Format
NTM	Neural Turing Machine

ONNX	Open Neural Network Exchange
OpenCL	Open Computing Language
OU	Ornstein-Uhlenbeck

PCA	Principal Component Analysis
PDF	Probability Density Function **or** Portable Document Format
PhD	Doctor of Philosophy
POMDP	Partially Observed Markov Decision Process
PReLU	Parametric Rectified Linear Unit
PSO	Particle Swarm Optimization

RADAM	Rectified ADAM
RAM	Random Access Memory
RBF	Radial Basis Function
RDPG	Recurrent Deterministic Policy Gradients
ReLU	Rectified Linear Unit
REM	Reflection Electron Microscopy
RHEED	Reflection high-energy electron diffraction
RHEELS	Reflection High Electron Energy Loss Spectroscopy
RL	Reinforcement Learning
RMLP	Recurrent Multilayer Perceptron
RMS	Root Mean Squared
RNN	Recurrent Neural Network
RReLU	Randomized Leaky Rectified Linear Unit
RTP	Research Technology Platform
RWI	Random Walk Initialization

SAE	Sparse Autoencoder
SELU	Scaled Exponential Linear Unit
SEM	Scanning Electron Microscopy
SGD	Stochastic Gradient Descent
SNE	Stochastic Neighbour Embedding
SNN	Self-Normalizing Neural Network
SPLEEM	Spin-Polarized Low-Energy Electron Microscopy
SSIM	Structural Similarity Index Measure
STM	Scanning Tunnelling Microscopy
SVD	Singular Value Decomposition
SVM	Support Vector Machine

TEM	Transmission Electron Microscopy
TIFF	Tag Image File Format
TPU	Tensor Processing Unit
tSNE	t-Distributed Stochastic Neighbour Embedding
TV	Total Variation
URL	Uniform Resource Locator
US-tSNE	Uniformly Separated t-Distributed Stochastic Neighbour Embedding
VAE	Variational Autoencoder
VAE-GAN	Variational Autoencoder Generative Adversarial Network
VBN	Virtual Batch Normalization
VGG	Visual Geometry Group
WDS	Wavelength Dispersive Spectroscopy
WEKA	Waikato Environment for Knowledge Analysis
WEMD	Warwick Electron Microscopy Datasets
WLEMD	Warwick Large Electron Microscopy Datasets
w.r.t.	With Respect To
XAI	Explainable Artificial Intelligence
XPS	X-Ray Photoelectron Spectroscopy
XRD	X-Ray Diffraction
XRF	X-Ray Fluorescence

List of Figures

Preface

Chapter 1 Review: Deep Learning in Electron Microscopy

Chapter 2 Warwick Electron Microscopy Datasets

Chapter 2 Supplementary Information: Warwick Electron Microscopy Datasets

Chapter 3 Adaptive Learning Rate Clipping Stabilizes Learning

training, and are less structured for adversarial training. Means, μ, and standard deviations, σ, of all pixels in each image are much higher for adversarial outputs. Enlarged 64×64 regions from the top left of each image are inset to ease comparison, and to show that systematic errors for MSE training are higher near output edges.

6. Test set root mean squared (RMS) intensity errors for spiral scans in $[0, 1]$ selected with binary masks. a) RMS errors decrease with increasing electron probe coverage, and are higher than deep learning supersampling (DLSS) errors. b) Frequency distributions of 20000 test set RMS errors for 100 bins in $[0, 0.224]$ and scan coverages in the legend.

Chapter 4 Supplementary Information: Partial Scanning Transmission Electron Microscopy with Deep Learning

S1. Discriminators examine random $w\times w$ crops to predict whether complete scans are real or generated. Generators are trained by multiple discriminators with different w. This figure was created with Inkscape.

S2. Two-stage generator that completes 512×512 micrographs from partial scans. A dashed line indicates that the same image is input to the inner and outer generator. Large scale features developed by the inner generator are locally enhanced by the outer generator and turned into images. An auxiliary trainer network restores images from inner generator features to provide direct feedback. This figure was created with Inkscape.

S3. Learning curves. a) Training with an auxiliary inner generator trainer stabilizes training, and converges to lower than two-stage training with fine tuning. b) Concatenating beam path information to inputs decreases losses. Adding symmetric residual connections between strided inner generator convolutions and transpositional convolutions increases losses. c) Increasing sizes of the first inner and outer generator convolutional kernels does not decrease losses. d) Losses are lower after more interations, and a learning rate (LR) of 0.0004; rather than 0.0002. Labels indicate inner generator iterations - outer generator iterations - fine tuning iterations, and k denotes multiplication by 1000 e) Adaptive learning rate clipped quartic validation losses have not diverged from training losses after 10^6 iterations. f) Losses are lower for outputs in [0, 1] than for outputs in [-1, 1] if leaky ReLU activation is applied to generator outputs.

Chapter 6 Improving Electron Micrograph Signal-to-Noise with an Atrous Convolutional Encoder-Decoder

Chapter 7 Exit Wavefunction Reconstruction from Single Transmission Electron Micrographs with Deep Learning

Chapter 7 Supplementary Information: Exit Wavefunction Reconstruction from Single Transmission Electron Micrographs with Deep Learning

S4. Input amplitudes, target phases and output phases of 224×224 multiple material training set wavefunctions for unseen flips, rotations and translations, and $n = 3$ simulation physics.

S5. Input amplitudes, target phases and output phases of 224×224 multiple material validation set wavefunctions for seen materials, unseen simulation hyperparameters, and $n = 3$ simulation physics.

S6. Input amplitudes, target phases and output phases of 224×224 multiple material validation set wavefunctions for unseen materials, unseen simulation hyperparameters are unseen, and $n = 3$ simulation physics.

S7. Input amplitudes, target phases and output phases of 224×224 validation set wavefunctions for restricted simulation hyperparameters, and $n = 3$ simulation physics.

S8. Input amplitudes, target phases and output phases of 224×224 validation set wavefunctions for restricted simulation hyperparameters, and $n = 3$ simulation physics.

S9. Input amplitudes, target phases and output phases of 224×224 $In_{1.7}K_2Se_8Sn_{2.28}$ training set wavefunctions for unseen flips, rotations and translations, and $n = 1$ simulation physics.

S10. Input amplitudes, target phases and output phases of 224×224 $In_{1.7}K_2Se_8Sn_{2.28}$ validation set wavefunctions for unseen simulation hyperparameters, and $n = 1$ simulation physics.

S11. Input amplitudes, target phases and output phases of 224×224 validation set wavefunctions for unseen simulation hyperparameters and materials, and $n = 1$ simulation physics. The generator was trained with $In_{1.7}K_2Se_8Sn_{2.28}$ wavefunctions.

S12. Input amplitudes, target phases and output phases of 224×224 $In_{1.7}K_2Se_8Sn_{2.28}$ training set wavefunctions for unseen flips, rotations and translations, and $n = 1$ simulation physics.

S13. Input amplitudes, target phases and output phases of 224×224 $In_{1.7}K_2Se_8Sn_{2.28}$ validation set wavefunctions for unseen simulation hyperparameters, and $n = 3$ simulation physics.

S14. Input amplitudes, target phases and output phases of 224×224 validation set wavefunctions for unseen simulation hyperparameters and materials, and $n = 3$ simulation physics. The generator was trained with $In_{1.7}K_2Se_8Sn_{2.28}$ wavefunctions.

List of Tables

Preface

Chapter 1 Review: Deep Learning in Electron Microscopy

Chapter 2 Warwick Electron Microscopy Datasets

Chapter 2 Supplementary Information: Warwick Electron Microscopy Datasets

Chapter 3 Adaptive Learning Rate Clipping Stabilizes Learning

Chapter 7 Exit Wavefunction Reconstruction from Single Transmission Electron Micrographs with Deep Learning

Acknowledgments

Most modern research builds on a high variety of intellectual contributions, many of which are often overlooked as there are too many to list. Examples include search engines, programming languages, machine learning frameworks, programming libraries, software development tools, computational hardware, operating systems, computing forums, research archives, and scholarly papers. To help developers with limited familiarity, useful resources for deep learning in electron microscopy are discussed in a review paper covered by ch. 1 of my thesis. For brevity, these acknowledgments will focus on personal contributions to my development as a researcher.

- Thanks go to Jeremy Sloan and Richard Beanland for supervision, internal peer review, and co-authorship.
- Thanks go to my Feedback Supervisors, Emma MacPherson and Jon Duffy, for comments needed to partially fulfil requirements of Doctoral Skills Modules (DSMs).
- I am grateful to Marin Alexe and Dong Jik Kim for supervising me during a summer project where I programmed various components of atomic force microscopes. It was when I first realized that I want to be a programmer. Before then, I only thought of programming as something that I did in my spare time.
- I am grateful to James Lloyd-Hughes for supervising me during a summer project where I automated Fourier analysis of ultrafast optical spectroscopy signals.
- I am grateful to my family for their love and support.

As a special note, I first taught myself machine learning by working through Mathematica documentation, implementing every machine learning example that I could find. The practice made use of spare time during a two-week course at the start of my Doctor of Philosophy (PhD) studentship, which was needed to partially fulfil requirements of the Midlands Physics Alliance Graduate School (MPAGS).

My Head of Department is David Leadley. My Director of Graduate Studies was Matthew Turner, then James Lloyd-Hughes after Matthew Turner retired.

I acknowledge funding from Engineering and Physical Sciences Research Council (EPSRC) grant EP/N035437/1 and EPSRC Studentship 1917382.

Declarations

This thesis is submitted to the University of Warwick in support of my application for the degree of Doctor of Philosophy. It has been composed by myself and has not been submitted in any previous application for any degree.

Parts of this thesis have been published by the author:

This thesis is typeset for physical printing and binding. In addition, the following version of this thesis[1] is typeset for online dissemination to improve readability.

> J. M. Ede. Advances in Electron Microscopy with Deep Learning. *arXiv preprint arXiv:2101.01178*, 2021

The following publications[2–9] are part of my thesis.

> J. M. Ede. Review: Deep Learning in Electron Microscopy. *arXiv preprint arXiv:2009.08328 (accepted by Machine Learning: Science and Technology – https://doi.org/10.1088/2632-2153/abd614)*, 2020
>
> J. M. Ede. Warwick Electron Microscopy Datasets. *Machine Learning: Science and Technology*, 1(4):045003, 2020
>
> J. M. Ede and R. Beanland. Adaptive Learning Rate Clipping Stabilizes Learning. *Machine Learning: Science and Technology*, 1:015011, 2020
>
> J. M. Ede and R. Beanland. Partial Scanning transmission Electron Microscopy with Deep Learning. *Scientific Reports*, 10(1):1–10, 2020
>
> J. M. Ede. Adaptive Partial Scanning Transmission Electron Microscopy with Reinforcement Learning. *arXiv preprint arXiv:2004.02786 (under review by Machine Learning: Science and Technology)*, 2020
>
> J. M. Ede and R. Beanland. Improving Electron Micrograph Signal-to-Noise with an Atrous Convolutional Encoder-Decoder. *Ultramicroscopy*, 202:18–25, 2019

J. M. Ede, J. J. P. Peters, J. Sloan, and R. Beanland. Exit Wavefunction Reconstruction from Single Transmission Electron Micrographs with Deep Learning. *arXiv preprint arXiv:2001.10938 (under review by Ultramicroscopy)*, 2020

J. M. Ede. Resume of Jeffrey Mark Ede. Zenodo, Online: https://doi.org/10.5281/zenodo.4429077, 2021

The following publications[10–13] are part of my thesis. However, they are appendices.

J. M. Ede. Supplementary Information: Warwick Electron Microscopy Datasets. Zenodo, Online: https://doi.org/10.5281/zenodo.3899740, 2020

J. M. Ede. Supplementary Information: Partial Scanning Transmission Electron Microscopy with Deep Learning. Online: https://static-content.springer.com/esm/art%3A10.1038%2Fs41598-020-65261-0/MediaObjects/41598_2020_65261_MOESM1_ESM.pdf, 2020

J. M. Ede. Supplementary Information: Adaptive Partial Scanning Transmission Electron Microscopy with Reinforcement Learning. Zenodo, Online: https://doi.org/10.5281/zenodo.4384708, 2020

J. M. Ede, J. J. P. Peters, J. Sloan, and R. Beanland. Supplementary Information: Exit Wavefunction Reconstruction from Single Transmission Electron Micrographs with Deep Learning. Zenodo, Online: https://doi.org/10.5281/zenodo.4277357, 2020

The following publications[14–26] are not part of my thesis. However, they are auxiliary to publications that are part of my thesis.

J. M. Ede. Warwick Electron Microscopy Datasets. *arXiv preprint arXiv:2003.01113*, 2020

J. M. Ede. Source Code for Warwick Electron Microscopy Datasets. Online: https://github.com/Jeffrey-Ede/datasets, 2020

J. M. Ede. Warwick Electron Microscopy Datasets Archive. Online: https://github.com/Jeffrey-Ede/datasets/wiki, 2020

J. M. Ede and R. Beanland. Adaptive Learning Rate Clipping Stabilizes Learning. *arXiv preprint arXiv:1906.09060*, 2019

J. M. Ede. Source Code for Adaptive Learning Rate Clipping Stabilizes Learning. Online: https://github.com/Jeffrey-Ede/ALRC, 2020

J. M. Ede and R. Beanland. Partial Scanning Transmission Electron Microscopy with Deep Learning. *arXiv preprint arXiv:1910.10467*, 2020

J. M. Ede. Deep Learning Supersampled Scanning Transmission Electron Microscopy. *arXiv preprint arXiv:1910.10467*, 2019

J. M. Ede. Source Code for Partial Scanning Transmission Electron Microscopy. Online: https://github.com/Jeffrey-Ede/partial-STEM, 2019

J. M. Ede. Source Code for Deep Learning Supersampled Scanning Transmission Electron Microscopy. Online: https://github.com/Jeffrey-Ede/DLSS-STEM, 2019

J. M. Ede. Source Code for Adaptive Partial Scanning Transmission Electron Microscopy with Reinforcement Learning. Online: https://github.com/Jeffrey-Ede/adaptive-scans, 2020

J. M. Ede. Improving Electron Micrograph Signal-to-Noise with an Atrous Convolutional Encoder-Decoder. *arXiv preprint arXiv:1807.11234*, 2018

J. M. Ede. Source Code for Improving Electron Micrograph Signal-to-Noise with an Atrous Convolutional Encoder-Decoder. Online: https://github.com/Jeffrey-Ede/Electron-Micrograph-Denoiser, 2019

J. M. Ede. Source Code for Exit Wavefunction Reconstruction from Single Transmission Electron Micrographs with Deep Learning. Online: https://github.com/Jeffrey-Ede/one-shot, 2019

The following publications[27–33] are not part of my thesis. However, they are referenced by my thesis, or are referenced by or associated with publications that are part of my thesis.

J. M. Ede. Progress Reports of Jeffrey Mark Ede: 0.5 Year Progress Report. Zenodo, Online: https://doi.org/10.5281/zenodo.4094750, 2020

J. M. Ede. Source Code for Beanland Atlas. Online: https://github.com/Jeffrey-Ede/Beanland-Atlas, 2018

J. M. Ede. Thesis Word Counting. Zenodo, Online: https://doi.org/10.5281/zenodo.4321429, 2020

J. M. Ede. Posters and Presentations. Zenodo, Online: https://doi.org/10.5281/zenodo.4041574, 2020

J. M. Ede. Autoencoders, Kernels, and Multilayer Perceptrons for Electron Micrograph Restoration and Compression. *arXiv preprint arXiv:1808.09916*, 2018

J. M. Ede. Source Code for Autoencoders, Kernels, and Multilayer Perceptrons for Electron Micrograph Restoration and Compression. Online: https://github.com/Jeffrey-Ede/Denoising-Kernels-MLPs-Autoencoders, 2018

J. M. Ede. Source Code for Simple Webserver. Online: https://github.com/Jeffrey-Ede/simple-webserver, 2019

All publications were produced during my period of study for the degree of Doctor of Philosophy in Physics at the University of Warwick.

The work presented (including data generated and data analysis) was carried out by the author except in the cases outlined below:

Chapter 1 Review: Deep Learning in Electron Microscopy

Jeremy Sloan and Martin Lotz internally reviewed my paper after I published it in the arXiv.

Chapter 2 Warwick Electron Microscopy Datasets

Richard Beanland internally reviewed my paper before it was published in the arXiv. Further, Jonathan Peters discussed categories used to showcase typical electron micrographs for readers with limited familiarity. At first, our datasets were openly accessible from my Google Cloud Storage. However, Richard Beanland contacted University of Warwick Information Technology Services to arrange for our datasets to also be openly accessible from University of Warwick data servers. Chris Parkin allocated server resources, advised me on data transfer, and handled administrative issues. In addition, datasets are openly accessible from Zenodo and my Google Drive.

Simulated datasets were created with clTEM multislice simulation software developed by a previous EM group PhD student, Mark Dyson, and maintained by a previous

EM group postdoctoral researcher, Jonathan Peters. Jonathan Peters advised me on processing data that I had curated from the Crystallography Open Database (COD) so that it could be input into clTEM simulations. Further, Jonathan Peters and I jointly prepared a script to automate multislice simulations. Finally, Jonathan Peters computed a third of our simulations on his graphical processing units (GPUs).

Experimental datasets were curated from University of Warwick Electron Microscopy (EM) Research Technology Platform (RTP) dataservers, and contain images collected by dozens of scientists working on hundreds of projects. Data was curated and published with permission of the Director of the EM RTP, Richard Beanland. In addition, data curation and publication were reviewed and approved by Research Data Officers, Yvonne Budden and Heather Lawler. I was introduced to the EM dataservers by Richard Beanland and Jonathan Peters, and my read and write access to the EM dataservers was set up by an EM RTP technician, Steve York.

Chapter 3 Adaptive Learning Rate Clipping Stabilizes Learning

Richard Beanland internally reviewed my paper after it was published in the arXiv. Martin Lotz later recommend the journal that I published it in. In addition, a Scholarly Communications Manager, Julie Robinson, advised me on finding publication venues and open access funding. I also discussed publication venues with editors of Machine Learning, Melissa Fearon and Peter Flach, and my Centre for Scientific Computing Director, David Quigley.

Chapter 4 Partial Scanning Transmission Electron Microscopy with Deep Learning

Richard Beanland internally reviewed an initial draft of my paper on partial scanning transmission electron microscopy (STEM). After I published our paper in the arXiv, Richard Beanland contributed most of the content in the first two paragraphs in the introduction of the journal paper. In addition, Richard Beanland and I both copyedited our paper.

Richard Beanland internally reviewed a paper on uniformly spaced scans after I published it in the arXiv. The uniformly spaced scans paper includes some experiments that we later combined into our partial STEM paper. Further, my experiments followed a preliminary investigation into compressed sensing with fixed randomly spaced masks, which Richard Beanland internally reviewed.

Chapter 5 Adaptive Partial Scanning Transmission Electron Microscopy with Reinforcement Learning

Jasmine Clayton, Abdul Mohammed, and Jeremy Sloan internally reviewed my paper after I published it in the arXiv.

Chapter 6 Improving Electron Micrograph Signal-to-Noise with an Atrous Convolutional Encoder-Decoder

After I published my paper in the arXiv, Richard Beanland internally reviewed it and advised that we publish it in a journal. In addition, Richard Beanland and I both copyedited our paper.

Chapter 7 Exit Wavefunction Reconstruction from Single Transmission Electron Micrographs with Deep Learning

Jeremy Sloan internally reviewed an initial draft of our paper. Afterwards, Jeremy Sloan contributed all crystal structure diagrams in our paper. The University of Warwick X-Ray Facility Manager, David Walker, suggested materials to showcase with their crystal structures, and a University of Warwick Research Fellow, Jessica Marshall, internally reviewed a figure showing exit wavefunction reconstructions (EWRs) with the crystal structures.

Richard Beanland contacted a professor at Humboldt University of Berlin, Christoph Koch, to ask for a DigitalMicrograph plugin, which I used to collect experimental focal series. Further, Richard Beanland helped me get started with focal series measurements, and internally reviewed some of my first focal series. In addition, Richard Beanland internally reviewed our paper.

Jonathan Peters drafted initial text about clTEM multislice simulations for a section of our paper on "Exit Wavefunction Datasets". In addition, Jonathan Peters internally reviewed our paper.

This thesis conforms to regulations governing the examination of higher degrees by research: The following regulations[34,35] were used during preparation of this thesis.

> Guide to Examinations for Higher Degrees by Research. University of Warwick Doctoral College, Online: https://warwick.ac.uk/services/dc/pgrassessments/gtehdr, 2020
>
> Regulation 38: Research Degrees. University of Warwick Calendar, Online: https://warwick.ac.uk/services/gov/calendar/section2/regulations/reg38pgr, 2020

The following guidance[36] was helpful during preparation of this thesis.

> Thesis Writing and Submission. University of Warwick Department of Physics, Online: https://warwick.ac.uk/fac/sci/physics/current/postgraduate/regs/thesis, 2020

The following thesis template[37] was helpful during preparation of this thesis.

> A Warwick Thesis Template. University of Warwick Department of Physics, Online: https://warwick.ac.uk/fac/sci/physics/staff/academic/mhadley/wthesis, 2020

Thesis structure and content was discussed with my previous Director of Graduate Studies, Matthew Turner, and my current Director of Graduate Studies, James Lloyd-Hughes, after Matthew Turner retired. My thesis was also discussed with my both my previous PhD supervisor, Richard Beanland, and my current PhD supervisor, Jeremy Sloan. My formal thesis plan was then reviewed and approved by both Jeremy Sloan and my feedback supervisor, Emma MacPherson. Finally, my complete thesis was internally reviewed by both Jeremy Sloan and Jasmine Clayton.

Permission is granted by the Chair of the Board of Graduate Studies, Colin Sparrow, for my thesis appendices to exceed length requirements usually set by the University of Warwick. This is in the understanding that my thesis appendices are not usually crucial to the understanding or examination of my thesis.

Research Training

This thesis presents a substantial original investigation of deep learning in electron microscopy. The only researcher in my research group or building with machine learning expertise was myself. This meant that I led the design, implementation, evaluation, and publication of experiments covered by my thesis. Where experiments were collaborative, I both proposed and led the collaboration.

Abstract

Following decades of exponential increases in computational capability and widespread data availability, deep learning is readily enabling new science and technology. This thesis starts with a review of deep learning in electron microscopy, which offers a practical perspective aimed at developers with limited familiarity. To help electron microscopists get started with started with deep learning, large new electron microscopy datasets are introduced for machine learning. Further, new approaches to variational autoencoding are introduced to embed datasets in low-dimensional latent spaces, which are used as the basis of electron microscopy search engines. Encodings are also used to investigate electron microscopy data visualization by t-distributed stochastic neighbour embedding. Neural networks that process large electron microscopy images may need to be trained with small batch sizes to fit them into computer memory. Consequently, adaptive learning rate clipping is introduced to prevent learning being destabilized by loss spikes associated with small batch sizes.

This thesis presents three applications of deep learning to electron microscopy. Firstly, electron beam exposure can damage some specimens, so generative adversarial networks were developed to complete realistic images from sparse spiral, gridlike, and uniformly spaced scans. Further, recurrent neural networks were trained by reinforcement learning to dynamically adapt sparse scans to specimens. Sparse scans can decrease electron beam exposure and scan time by 10-100× with minimal information loss. Secondly, a large encoder-decoder was developed to improve transmission electron micrograph signal-to-noise. Thirdly, conditional generative adversarial networks were developed to recover exit wavefunction phases from single images. Phase recovery with deep learning overcomes existing limitations as it is suitable for live applications and does not require microscope modification. To encourage further investigation, scientific publications and their source files, source code, pretrained models, datasets, and other research outputs covered by this thesis are openly accessible.

Preface

This thesis covers a subset of my scientific papers on advances in electron microscopy with deep learning. The papers were prepared while I was a PhD student at the University of Warwick in support of my application for the degree of PhD in Physics. This thesis reflects on my research, unifies covered publications, and discusses future research directions. My papers are available as part of chapters of this thesis, or from their original publication venues with hypertext and other enhancements. This preface covers my initial motivation to investigate deep learning in electron microscopy, structure and content of my thesis, and relationships between included publications. Traditionally, physics PhD theses submitted to the University of Warwick are formatted for physical printing and binding. However, I have also formatted a copy of my thesis for online dissemination to improve readability[1].

I Initial Motivation

When I started my PhD in October 2017, we were unsure if or how machine learning could be applied to electron microscopy. My PhD was funded by EPSRC Studentship 1917382[38] titled "Application of Novel Computing and Data Analysis Methods in Electron Microscopy", which is associated with EPSRC grant EP/N035437/1[39] titled "ADEPT – Advanced Devices by ElectroPlaTing". As part of the grant, our initial plan was for me to spend a couple of days per week using electron microscopes to analyse specimens sent to the University of Warwick from the University of Southampton, and to invest remaining time developing new computational techniques to help with analysis. However, an additional scientist was not needed to analyse specimens, so it was difficult for me to get electron microscopy training. While waiting for training, I was tasked with automating analysis of digital large angle convergent beam electron diffraction[40] (D-LACBED) patterns. However, we did not have a compelling use case for my D-LACBED software[27,41]. Further, a more senior PhD student at the University of Warwick, Alexander Hubert, was already investigating convergent beam electron diffraction[40,42] (CBED).

My first machine learning research began five months after I started my PhD. Without a clear research direction or specimens to study, I decided to develop artificial neural networks (ANNs) to generate artwork. My dubious plan was to create image processing pipelines for the artwork, which I would replace with electron micrographs when I got specimens to study. However, after investigating artwork generation with randomly initialized multilayer perceptrons[43,44], then by style transfer[45,46], and then by fast style transfer[47], there were still no specimens for me to study. Subsequently, I was inspired by NVIDIA's research on semantic segmentation[48] to investigate semantic segmentation with DeepLabv3+[49]. However, I decided that it was unrealistic for me to label a large new electron microscopy dataset for semantic segmentation by myself. Fortunately, I had read about using deep neural networks (DNNs) to reduce image compression artefacts[50], so I wondered if a similar approach based on DeepLabv3+ could improve electron micrograph signal-to-noise. Encouragingly, it would not require time-consuming image labelling. Following a successful investigation into improving signal-to-noise, my first scientific paper[7] (ch. 6) was submitted a few months later, and my experience with deep learning enabled subsequent investigations.

II Thesis Structure

An overview of the first seven chapters in this thesis is presented in fig. 1. The first chapter is introductory and covers a review of deep learning in electron microscopy, which offers a practical perspective aimed at developers with limited familiarity. The next two chapters are ancillary and cover new datasets and an optimization algorithm used in later chapters. The final four chapters before conclusions cover investigations of deep learning in electron microscopy. Each of the first seven chapter covers a combination of journal papers, preprints, and ancillary outputs such as source code, datasets, and pretrained models, and supplementary information.

At the University of Warwick, physics PhD theses that cover publications[51,52] are unusual. Instead, most theses are scientific monographs. However, declining impact of monographic theses is long-established[53], and I felt that scientific publishing would push me to produce higher-quality research. Moreover, I think that publishing is an essential part of scientific investigation, and external peer reviews[54–58] often helped me to improve my papers. Open access to PhD theses increases visibility[59,60] and enables their use as data mining resources[60,61], so digital copies of physics PhD theses are archived by the University of Warwick[62]. However, archived theses are

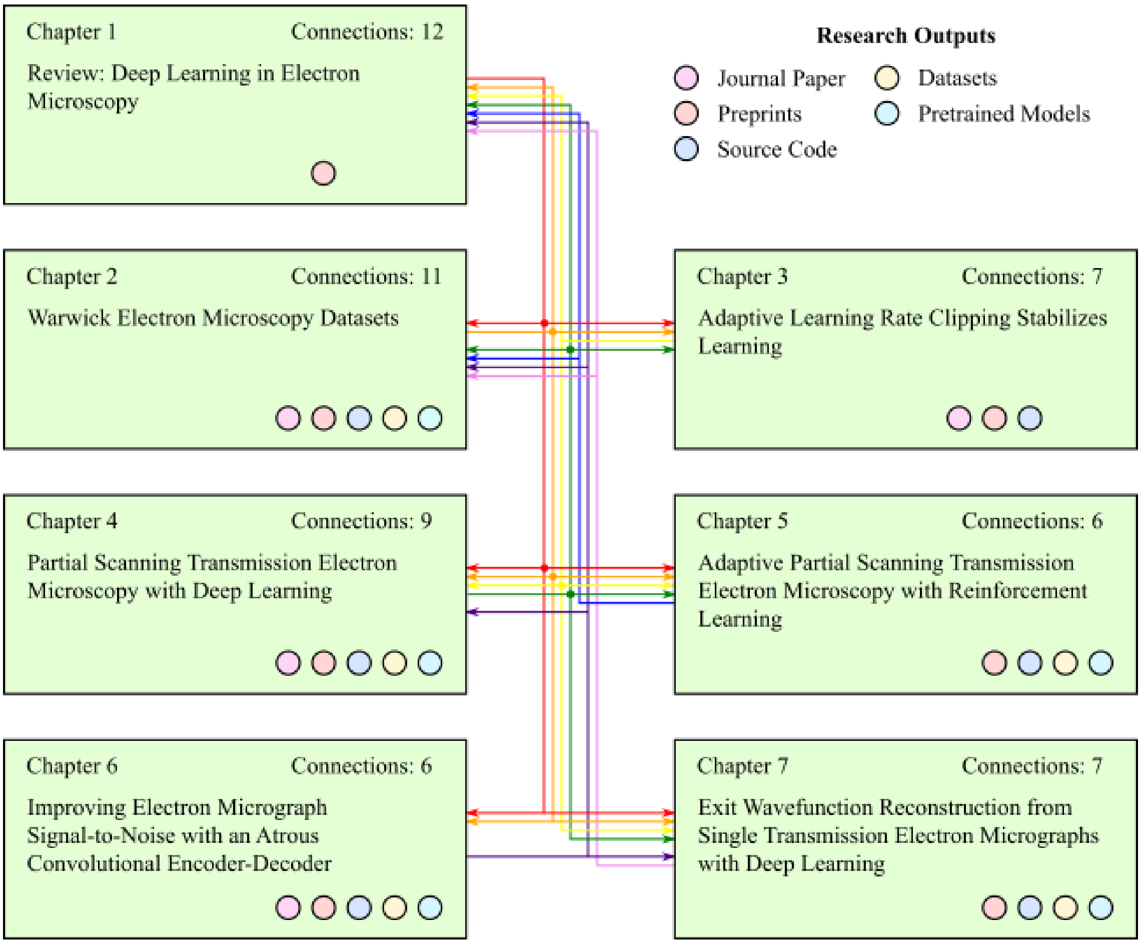
Research Outputs
Journal Paper
Datasets
Preprints
Pretrained Models
Source Code
Chapter 1
Connections: 12
Review: Deep Learning in Electron Microscopy
Chapter 2
Connections: 11
Warwick Electron Microscopy Datasets
Chapter 3
Connections: 7
Adaptive Learning Rate Clipping Stabilizes Learning
Chapter 4
Connections: 9
Partial Scanning Transmission Electron Microscopy with Deep Learning
Chapter 5
Connections: 6
Adaptive Partial Scanning Transmission Electron Microscopy with Reinforcement Learning
Chapter 6
Connections: 6
Improving Electron Micrograph Signal-to-Noise with an Atrous Convolutional Encoder-Decoder
Chapter 7
Connections: 7
Exit Wavefunction Reconstruction from Single Transmission Electron Micrographs with Deep Learning

Figure 1: Connections between publications covered by chapters of this thesis. An arrow from chapter x to chapter y indicates that results covered by chapter y depend on results covered by chapter x. Labels indicate types of research outputs associated with each chapter, and total connections to and from chapters.

usually formatted for physical printing and binding. To improve readability, I have also formatted a copy of my thesis for online dissemination[1], which is published in the arXiv[63,64] with its Latex[65–67] source files.

All my papers were first published as arXiv preprints under Creative Commons Attribution 4.0[68] licenses, then submitted to journals. As discussed in my review[2] (ch. 1), advantages of preprint archives[69–71] include ensuring that research is openly accessible[72], increasing discovery and citations[73–77], inviting timely scientific discussion, and raising awareness to reduce unnecessary duplication of research. Empirically, there are no significant textual differences between arXiv preprints and corresponding journal papers[78]. However, journal papers appear to be slightly higher quality than biomedical preprints[78,79], suggesting that formatting and copyediting practices vary between scientific disciplines. Overall, I think that a lack of differences between journal papers and preprints may be a result of publishers separating language editing into premium services[80–83], rather than including extensive language editing in their usual publication processes. Increasing textual quality is correlated with increasing likelihood that an article will be published[84]. However, most authors appear to be performing copyediting themselves to avoid extra fees.

A secondary benefit of posting arXiv preprints is that their metadata, an article in portable document format[85,86] (PDF), and any Latex source files are openly accessible. This makes arXiv files easy to reuse, especially if they are published under permissive licenses[87]. For example, open accessibility enabled arXiv files to be curated into a large dataset[88] that was used to predict future research trends[89]. Further, although there is no requirement for preprints to peer reviewed, preprints can enable early access to papers that have been peer reviewed. As a case in point, all preprints covered by my thesis have been peer reviewed. Further, the arXiv implicitly supports peer review by providing contact details of authors, and I have both given and received feedback about arXiv papers. In addition, open peer review platforms[90], such as OpenReview[91,92], can be used to explicitly seek peer review. There is also interest in integrating peer review with the arXiv, so a conceptual peer review model has been proposed[93].

Description	Words in Text	Words in Figures	Words in Algorithms	Total Words
Review paper in chapter 1	15156	2680	74	17910
Ancillary paper in chapter 2	4243	1360	0	5603
Ancillary paper in chapter 3	2448	680	344	3472
Paper in chapter 4	3864	1300	0	5164
Paper in chapter 5	3399	900	440	4739
Paper in chapter 6	2933	1100	0	4033
Paper in chapter 7	4396	1240	0	5636
Remainder of the thesis	7950	280	0	8230
Complete thesis	**44389**	**9540**	**858**	**54787**

Table 1: Word counts for papers included in thesis chapters, the remainder of the thesis, and the complete thesis.

This thesis covers a selection of my interconnected scientific papers. Word counts for my papers and covering text are tabulated in table 1. Figures are included in word counts by adding products of nominal word densities and figure areas. However, acknowledgements, references, tables, supplementary information, and similar contents are not included as they do not count towards my thesis length limit of 70000 words. For details, notes on my word counting procedure are openly accessible[29]. Associated research outputs, such as source code and datasets, are not directly included in my thesis due to format restrictions. Nevertheless, my source code is openly accessible from GitHub[94], and archived releases of my source code are openly accessible from

Zenodo[95]. In addition, links to openly accessible pretrained models are provided in my source code documentation. Finally, links to openly accessible datasets are in my papers, source code documentation, and datasets paper[3] (ch. 2).

III Connections

Connections between publications covered by my thesis are shown in fig. 1. The most connected chapter covers my review paper[2] (ch. 1). All my papers are connected to my review paper as literature reviews informed their introductions, methodologies, and discussions. My review paper also discusses and builds upon the results of my earlier publications. For example, images published in my earlier papers are reused in my review paper to showcase applications of deep learning in electron microscopy. In addition, my review paper covers Warwick Electron Microscopy Datasets[3] (WEMD, ch. 2), adaptive learning rate clipping[4] (ALRC, ch. 3), sparse scans for compressed sensing in STEM[5] (ch. 4), improving electron microscope signal-to-noise[7] (ch. 6), and EWR[8] (ch. 7). Finally, compressed sensing with dynamic scan paths that adapt to specimens[6] (ch. 5) motivated my review paper sections on recurrent neural networks (RNNs) and reinforcement learning (RL).

The second most connected chapter, ch. 2, is ancillary and covers WEMD[3], which include large new datasets of experimental transmission electron microscopy (TEM) images, experimental STEM images, and simulated exit wavefunctions. The TEM images were curated to train an ANN to improve signal-to-noise[7] (ch. 6) and motivated the proposition of a new approach to EWR[8] (ch. 7). The STEM images were curated to train ANNs for compressed sensing[5] (ch. 4). Training our ANNs with full-size images was impractical with our limited computational resources, so I created dataset variants containing 512×512 crops from full-size images for both the TEM and STEM datasets. However, 512×512 STEM crops were too large to efficiently train RNNs to adapt scan paths[6] (ch. 5), so I also created 96×96 variants of datasets for rapid initial development. Finally, datasets of exit wavefunctions were simulated as part of our initial investigation into EWR from single TEM images with deep learning[8] (ch. 7).

The other ancillary chapter, ch. 3, covers ALRC[4], which was originally published as an appendix in the first version of our partial STEM preprint[19] (ch. 4). The algorithm was developed to stabilize learning of ANNs being developed for partial STEM, which were destabilized by loss spikes when training with a batch size of 1. My aim was to make experiments[11] easier to compare by preventing learning destabilized by large loss spikes from complicating comparisons. However, ALRC was so effective that I continued to investigate it, increasing the size of the partial STEM appendix. Eventually, the appendix became so large that I decided to turn it into a short paper. To stabilize training with small batch sizes, ALRC was also applied to ANN training for uniformly spaced scans[5,20] (ch. 4). In addition, ALRC inspired adaptive loss clipping to stabilize RNN training for adaptive scans[6] (ch. 5). Finally, I investigated applying ALRC to ANN training for EWR[8] (ch. 7). However, ALRC did not improve EWR as training with a batch size of 32 was not destabilized by loss spikes.

My experiments with compressed sensing showed that ANN performance varies for different scan paths[5] (ch. 4). This motivated the investigation of scan shapes that adapt to specimens as they are scanned[6] (ch. 5). I had found that ANNs for TEM denoising[7] (ch. 6) and uniformly spaced sparse scan completion[20] exhibit significant structured systematic error variation, where errors are higher near output edges. Subsequently, I investigated average partial STEM output errors and found that errors increase with increasing distance from scan paths[5] (ch. 4). In part, structured systematic error variation in partial STEM[5] (ch. 4) motivated my investigation of adaptive scans[6] (ch. 5) as I reasoned that being able to more closely scan regions where errors would otherwise be highest could decrease mean errors.

Most of my publications are connected by their source code as it was partially reused in successive experiments. Source code includes scripts to develop ANNs, plot graphs, create images for papers, and typeset with Latex. Following my publication chronology, I partially reused source code created to improve signal-to-noise[7] (ch. 6) for partial STEM[5] (ch. 4). My partial STEM source code was then partially reused for my other investigations. Many of my publications are also connected because datasets curated for my first investigations were reused in my later investigations. For example, improving signal-to-noise[7] (ch. 6) is connected to EWR[8] (ch. 7) as the availability of my large dataset of TEM images prompted the proposition of, and may enable, a new approach to EWR. Similarly, partial STEM[5] (ch. 4) is connected to adaptive scans[6] (ch. 5) as my large dataset of STEM images was used to derive smaller datasets used to rapidly develop adaptive scan systems.

Chapter 1

Review: Deep Learning in Electron Microscopy

1.1 Scientific Paper

This chapter covers the following paper[2].

> J. M. Ede. Review: Deep Learning in Electron Microscopy. *arXiv preprint arXiv:2009.08328 (accepted by Machine Learning: Science and Technology –* `https://doi.org/10.1088/2632-2153/abd614`*)*, 2020

1.2 Reflection

This introductory chapter covers my review paper[96] titled “Review: Deep Learning in Electron Microscopy”[2]. It is the first in-depth review of deep learning in electron microscopy and offers a practical perspective that is aimed at developers with limited familiarity. My review was crafted to be covered by the introductory chapter of my PhD thesis, so focus is placed on my research methodology. Going through its sections in order of appearance, “Introduction” covers and showcases my earlier research, “Resources” introduces resources that enabled my research, “Electron Microscopy” covers how I simulated exit wavefunctions and integrated ANNs with electron microscopes, “Components” introduces functions used to construct my ANNs, “Architecture” details ANN archetypes used in my research, “Optimization” covers how my ANNs were trained, and “Discussion” offers my perspective on deep learning in electron microscopy.

There are many review papers on deep learning. Some reviews of deep learning focus on computer science[97–101], whereas others focus on specific applications such as computational imaging[102], materials science[103–105], and the physical sciences[106]. As a result, I anticipated that another author might review deep learning in electron microscopy. To avoid my review being easily surpassed, I leveraged my experience to offer practical perspectives and comparative discussions to address common causes of confusion. In addition, content is justified by extensive references to make it easy to use as a starting point for future research. Finally, I was concerned that information about how to get started with deep learning in electron microscopy was fragmented and unclear to unfamiliar developers. This was often problematic when I was asked about getting started with machine learning, and I was especially conscious of it as my friend, Rajesh Patel, asked me for advice when I started writing my review. Consequently, I included a section that introduces useful resources for deep learning in electron microscopy.

Chapter 2

Warwick Electron Microscopy Datasets

2.1 Scientific Paper

This paper covers the following paper[3] and its supplementary information[10].

> J. M. Ede. Warwick Electron Microscopy Datasets. *Machine Learning: Science and Technology*, 1(4):045003, 2020
>
> J. M. Ede. Supplementary Information: Warwick Electron Microscopy Datasets. Zenodo, Online: https://doi.org/10.5281/zenodo.3899740, 2020

2.2 Amendments and Corrections

There are amendments or corrections to the paper[3] covered by this chapter.

Location: Page 4, caption of fig. 2.
Change: "...at 500 randomly selected images..." should say "...at 500 randomly selected data points...".

2.3 Reflection

This ancillary chapter covers my paper titled "Warwick Electron Microscopy Datasets"[3] and associated research outputs[10,14–16]. My paper presents visualizations for large new electron microscopy datasets published with our earlier papers. There are 17266 TEM images curated to train our denoiser[7] (ch. 6), 98340 STEM images curated to train generative adversarial networks (GANs) for compressed sensing[5,20] (ch. 4), and 98340 TEM exit wavefunctions simulated to investigate EWR[8] (ch. 7), and derived datasets containing smaller TEM and STEM images that I created to rapidly prototype of ANNs for adaptive partial STEM[6] (ch. 5). To improve visualizations, I developed new regularization mechanisms for variational autoencoders[107–109] (VAEs), which were trained to embed high-dimensional electron micrographs in low-dimensional latent spaces. In addition, I demonstrate that VAEs can be used as the basis of electron micrograph search engines. Finally, I provide extensions to t-distributed stochastic neighbour embedding[110–114] (tSNE) and interactive dataset visualizations.

Making our large machine learning datasets openly accessible enables our research to be reproduced[115], standardization of performance comparisons, and dataset reuse in future research. Dissemination of large datasets is enabled by the internet[116,117], for example, through fibre optic[118] broadband[119,120] or satellite[121,122] connections. Subsequently, there are millions of open access datasets[123,124] that can be used for machine learning[125,126]. Performance of ANNs usually increases with increasing training dataset size[125], so some machine learning datasets have millions of examples. Examples of datasets with millions of examples include DeepMind Kinetics[127], ImageNet[128], and YouTube 8M[129]. Nevertheless, our datasets containing tens of thousands of examples are more than sufficient for initial exploration of deep learning in electron microscopy. For reference, some datasets used for initial explorations of deep learning for Coronavirus Disease 2019[130–132] (COVID-19) diagnosis are 10× smaller[133] than WEMD.

There are many data clustering algorithms[134–140] that can group data for visualization. However, tSNE is a *de facto* default as it often outperforms other algorithms[110]. For context, tSNE is a variant of stochastic neighbour embedding[141] (SNE) where a heavy-tailed Student's t-distribution is used to measure distances between embedded data points. Applications of tSNE include bioinformatics[142,143], forensic science[144,145], medical signal processing[146–148], particle physics[149,150], smart electricity metering[151], and sound synthesis[152]. Before tSNE, data is often embedded in a low-dimensional space to reduce computation, suppress noise, and prevent Euclidean distances used in tSNE optimization being afflicted by the curse of dimensionality[153]. For example, the original tSNE paper suggests using principal component analysis[154–157] (PCA) to reduce data dimensionality to 30 before applying tSNE[110].

Extensions of tSNE can improve clustering. For example, graphical processing unit accelerated implementations of tSNE[158,159] can speedup clustering 50-700×. Alternatively, approximate tSNE[160] (A-tSNE) can trade accuracy for decreased computation time. Our tSNE visualizations took a couple of hours to optimize on an Intel i7-6700 central processing unit (CPU) as we used 10000 iterations to ensure that clusters stabilized. It follows that accelerated tSNE implementations may be preferable to reduce computation time. Another extension is to adjust distances used for tSNE optimization with a power transform based on the intrinsic dimension of each point. This can alleviate the curse of dimensionality for high-dimensional data[153]; however, it was not necessary for our data as I used VAEs to reduce image dimensionality to 64 before tSNE. Finally, tSNE early exaggeration (EE), where probabilities modelling distances in a high-dimensional space are increased, and number of iterations can be automatically tuned with opt-tSNE[161]. Tuning can significantly improve visualizations, especially for large datasets with millions of examples. However, I doubt that opt-tSNE would result in large improvements to clustering as our datasets contain tens of thousands of examples, where tSNE is effective. Nevertheless, I expect that opt-tSNE could have improved clustering if I had been aware of it.

Further extensions to tSNE are proposed in my paper[3,10]. I think that the most useful extension uniformly separates clustered points based clustering density. Uniformly separated tSNE (US-tSNE) can often double whitespace utilization, which could make tSNE visualizations more suitable for journals, websites, and other media where space is limited. However, the increased whitespace utilization comes at the cost of removing information about the structure of clusters. Further, my preliminary implementation of US-tSNE is limited insofar that Clough-Tocher cubic Bezier interpolation[162] used to map tSNE points to a uniform map is only applied to points within their convex hull. I also proposed a tSNE extension that uses standard deviations encoded by VAEs to inform clustering as this appeared to slightly improve clustering. However, I later found that using standard deviations appears to decrease similarity of nearest neighbours in tSNE visualizations. As a result, I think that how extra information encoded in standard deviations is used to inform clustering may merit further investigation.

To improve VAE encodings for tSNE, I applied a variant of batch normalization to their latent spaces. This avoids needing to tune a hyperparameter to balance VAE decoder and Kullback-Leibler (KL) losses, which is architecture-specific and can be complicated by relative sizes of their gradients varying throughout training. I also considered adaptive gradient balancing[163,164] of losses; however, that would require separate backpropagation through the VAE generator for each loss, increasing computation. To increase image realism, I added Sobel losses to mean squared errors (MSEs). Sobel losses often improve realism as human vision is sensitive to edges[165]. In addition, Sobel losses require less computation than VAE training with GAN[166] or perceptual[167] losses. Another computationally inexpensive approach to improve generated image realism is to train with structural similarity index measures[168] (SSIMs) instead of MSEs[169].

My VAEs are used as the basis of my openly accessible electron microscopy search engines. I observe that top-5 search results are usually successful insofar that they contain images that are similar to input images. However, they often contain some images that are not similar, possibly due to there not being many similar images in our datasets. Thus, I expect that search results could be improved by increasing dataset size. Increasing input image size from 96×96 to a couple of hundred pixels and increasing training iterations could also improve performance. Further, training could be modified to encode binary latent variables for efficient hashing[170–175]. Finally, I think that an interesting research direction is to create a web interface for an electron microscopy search engine that indexes institutional electron microscopy data servers. Such a search engine could enhance collaboration by making it easier to find electron microscopists working on interesting projects.

An application of my VAEs that is omitted from my paper is that VAE generators could function as portable electron microscopy image generators. For example, to create training data for machine learning. For comparison, my VAE generators require roughly 0.1% of the storage space needed for my image datasets to store their trainable parameters. However, I was concerned that a distribution of generated images might be biased by catastrophic forgetting[176]. Further, a distribution of generated images could be sensitive to ANN architecture and learning policy, including when training is stopped[177,178]. Nevertheless, I expect that data generated from by VAEs could be used for pretraining to improve ANN robustness[179]. Overall, I think it will become increasingly practical to use VAEs or GANs as high-quality data generators as ANN architectures and learning policies are improved.

Perhaps the main limitation of my paper is that I did not introduce my preferred abbreviation, "WEMD", for "Warwick Electron Microscopy Datasets". Further, I did not define "WEMD" in my WEMD preprint[14]. Subsequently, I introduced my preferred abbreviation in my review of deep learning in electron microscopy[2] (ch. 1). I also defined an abbreviation, "WLEMD", for "Warwick Large Electron Microscopy Datasets" in the first version of the partial STEM preprint[19] (ch. 4). Another limitation is that my paper only details datasets that had already been published, or that were derived from the published datasets. For example, Richard Beanland and I successfully co-authored an application for funding to simulate tens of thousands of CBED patterns with Felix[180], which are not detailed in my paper. The CBED dataset requires a couple of terabytes of storage and has not been processed for dissemination. Nevertheless, Richard Beanland[1] may be able to provide the CBED dataset upon request.

[1] Email: [redacted]

Chapter 3

Adaptive Learning Rate Clipping Stabilizes Learning

3.1 Scientific Paper

This chapter covers the following paper[4].

> J. M. Ede and R. Beanland. Adaptive Learning Rate Clipping Stabilizes Learning. *Machine Learning: Science and Technology*, 1:015011, 2020

3.2 Amendments and Corrections

There are amendments or corrections to the paper[4] covered by this chapter.

Location: Page 3, image in fig. 1.
Change: A title above the top two graphs is cut off. The missing title said "With Adaptive Learning Rate Clipping", and is visible in our preprint[17].

Location: Last paragraph starting on page 7.
Change: "...inexpensive alternative to gradient clipping in high batch size training where..." should say "...inexpensive alternative to gradient clipping where...".

3.3 Reflection

This ancillary chapter covers my paper titled "Adaptive Learning Rate Clipping Stabilizes Learning"[4] and associated research outputs[17,18]. The ALRC algorithm was developed to prevent loss spikes destabilizing training of DNNs for partial STEM[5] (ch. 4). To fit the partial STEM ANN in GPU memory, it was trained with a batch size of 1. However, using a small batch size results in occasional loss spikes, which meant that it was sometimes necessary to repeat training to compare performance with earlier experiments where learning had not been destabilized by loss spikes. I expected that I could adjust training hyperparameters to stabilize learning; however, I had optimized the hyperparameters and training was usually fine. Thus, I developed ALRC to prevent loss spikes from destabilizing learning. Initially, ALRC was included as an appendix in the first version of the partial STEM preprint[19]. However, ALRC was so effective that I continued to investigate. Eventually, there were too many ALRC experiments to comfortably fit in an appendix of the partial STEM paper, so I separated ALRC into its own paper.

There are variety of alternatives to ALRC that can stabilize learning. A popular alternative is training with Huberized losses[181,182],

$$\text{Huber}(L) = \min(L, (\lambda L)^{1/2})\,, \tag{3.1}$$

where L is a loss and λ is a training hyperparameter. However, I found that Huberized learning continued to be destabilized by loss spikes. I also considered gradient clipping[183–185]. However, my DNNs for partial STEM have many millions of trainable parameters, so computational requirements for gradient clipping are millions of times higher than applying ALRC to losses. Similarly, rectified ADAM[186] (RADAM), can stabilize learning by decreasing trainable parameter learning rates if adaptive learning rates of an ADAM[187] optimizer have high variance. However, computational requirements of RADAM are also often millions of times higher than ALRC as RADAM adapts adaptive learning rates for every trainable parameter.

Overall, I think that ALRC merits further investigation. ALRC is computationally inexpensive, can be applied to any loss function, and appears to either stabilize learning or have no significant effect. Further, ALRC can often readily improve ANN training that would otherwise be destabilized loss spikes. However, I suspect that ALRC may slightly decrease performance where learning is not destabilized by loss spikes as ALRC modifies training losses. In addition, I have only investigated applications of ALRC to mean square and quartic errors per training example of deep convolutional neural networks (CNNs). Applying ALRC to losses for individual pixels of CNN outputs or to losses at each step of a recurrent neural network (RNN) may further improve performance. Encouragingly, my initial experiments with ALRC variants[4] show that a variety approaches improve training that would otherwise be destabilized by loss spikes.

Chapter 4

Partial Scanning Transmission Electron Microscopy with Deep Learning

4.1 Scientific Paper

This chapter covers the following paper[5] and its supplementary information[11].

> J. M. Ede and R. Beanland. Partial Scanning transmission Electron Microscopy with Deep Learning. *Scientific Reports*, 10(1):1–10, 2020
>
> J. M. Ede. Supplementary Information: Partial Scanning Transmission Electron Microscopy with Deep Learning. Online: https://static-content.springer.com/esm/art%3A10.1038%2Fs41598-020-65261-0/MediaObjects/41598_2020_65261_MOESM1_ESM.pdf, 2020

4.2 Amendments and Corrections

There are amendments or corrections to the paper[5] covered by this chapter.

Location: Reference 13 in the bibliography.
Change: "Sang, X. *et al*. Dynamic Scan Control in STEM: Spiral Scans. Adv. Struct. Chem. Imaging 2, 6 (2017)" should say "Sang, X. *et al*. Dynamic Scan Control in STEM: Spiral Scans. Adv. Struct. Chem. Imaging 2, 1–8 (2016)".

4.3 Reflection

This chapter covers our paper titled "Partial Scanning Transmission Electron Microscopy with Deep Learning"[5] and associated research outputs[11,16,19–22,188], which were summarized by Bethany Connolly[189]. Our paper presents some of my investigations into compressed sensing of STEM images. Specifically, it combines results from two of my arXiv papers about compressed sensing with contiguous paths[19] and uniformly spaced grids[20] of probing locations. A third investigation into compressed sensing with a fixed random grid of probing locations was not published as I think that uniformly spaced grid scans are easier to implement on most scan systems. Further, reconstruction errors were usually similar for uniformly spaced and fixed random grids with the same coverage. Nevertheless, a paper I drafted on fixed random grids is openly accessible[190]. Overall, I think that compressed sensing with DNNs is a promising approach to reduce electron beam damage and scan time by 10-100× with minimal information loss.

My comparison of spiral and uniformly spaced grid scans with the same ANN architecture, learning policy and training data indicates that errors are lower for uniformly spaced grids. However, the comparison is not conclusive as ANNs were trained for a few days, rather than until validation errors plateaued. Further, a fair comparison is difficult as suitability of architectures and learning policies may vary for different scan paths. Higher performance of uniformly spaced grids can be explained by content at the focus of most electron micrographs being imaged at 5-10× its Nyquist rate[3] (ch. 2). It follows that high-frequency information that is accessible from neighbouring pixels in contiguous scans is often almost redundant. Overall, I think the best approach may combine both contiguous and uniform spaced grid scans. For example, a contiguous scan ANN could exploit high-frequency information to complete an image, which could then be mapped to a higher resolution image by an ANN for uniformly spaced scans. Indeed, functionality for contiguous and uniformly spaced grid scans could be combined into a single ANN.

Most STEM scan systems can raster uniformly spaced grids of probing locations. However, scan systems often have to be modified to perform spiral or other custom scans[191,192]. Modification is not difficult for skilled programmers. For example, Jonathan Peters[1] created a custom scan controller prototype based on my field programmable gate array[193] (FPGA) within one day. Custom scans are often more distorted than raster scans. However, distortions can be minimized by careful choice

[1] Email: [redacted]

of custom scan speed and path shape[191]. Alternatively, ANNs can correct electron microscope scan distortions[194,195]. We planned to use my FPGA to develop an openly accessible custom scan controller near the end of my PhD; however, progress was stalled by COVID-19 national lockdowns in the United Kingdom[196]. As a result, I invested time that we had planned to use for FPGA deployment to review deep learning in electron microscopy[2] (ch. 1).

To complete realistic images, generators were trained with MSEs or as part of GANs. However, GANs can introduce uncertainty into scientific investigation as they can generate realistic outputs, even if scan coverage is too low to reliably complete a region[5]. Consequently, I investigated reducing uncertainty by adapting scan coverage[6] to imaging regions (ch. 5). Alternatively, there are a variety of methods to quantify DNN uncertainty[197–203]. For example, uncertainty can be predicted by ANNs[204,205], Bayesian uncertainty approximation[206–209], or from variance of bootstrap aggregated[210] (bagged) model outputs. To address uncertainty, we present mean errors for 20000 test images, showing that errors are higher further away from scan paths. However, we do not provide an approach to quantify uncertainty of individual images, which could be critical to make scientific conclusions. Overall, I think that further investigation of uncertainty may be necessary before DNNs are integrated into default operating configurations of electron microscopes.

A GAN could learn to generate any realistic STEM images, rather than outputs that correspond to inputs. To train GANs to generate outputs that correspond to inputs, I added MSEs between blurred input and output images to generator losses. Blurring prevented MSEs from strongly suppressing high-frequency noise characteristics. I also investigated adding distances between features output by discriminator layers for real and generated images to generator losses[48]. However, feature distances require more computation than MSEs, and both feature distances and MSEs result in similar SSIMs[190] between completed and true scans. As a result, I do not think that other computationally inexpensive additional losses, such as SSIMs or mean absolute errors, would substantially improve performance. Finally, I considered training generators to minimize perceptual losses[211]. However, most pretrained models used for feature extraction are not trained on electron micrographs or scientific images. Consequently, I was concerned that pretrained models might not clearly perceive characteristics specific to electron micrographs, such as noise.

Chapter 5

Adaptive Partial Scanning Transmission Electron Microscopy with Reinforcement Learning

5.1 Scientific Paper

This chapter covers the following paper[6] and its supplementary information[12].

> J. M. Ede. Adaptive Partial Scanning Transmission Electron Microscopy with Reinforcement Learning. *arXiv preprint arXiv:2004.02786 (under review by Machine Learning: Science and Technology)*, 2020
>
> J. M. Ede. Supplementary Information: Adaptive Partial Scanning Transmission Electron Microscopy with Reinforcement Learning. Zenodo, Online: https://doi.org/10.5281/zenodo.4384708, 2020

5.2 Reflection

This chapter covers my paper titled "Adaptive Partial Scanning Transmission Electron Microscopy with Reinforcement Learning"[6] and associated research outputs[16,23]. It presents an initial investigation into STEM compressed sensing with contiguous scans that are piecewise adapted to specimens. Adaptive scanning is a finite-horizon partially observed Markov decision process[212,213] (POMDP) with continuous actions and sparse rewards: Scan directions are chosen at each step based on previously observed path segments and a sparse reward is given by correctness completed sparse scans. Scan directions are decided by an actor RNN that cooperates with a generator CNN that completes full scans from sparse scans. Generator losses are not differentiable with respect to actor actions, so I introduced a differentiable critic RNN to predict generator losses from actor actions and observations. The actor and critic are trained by reinforcement learning with a new extension of recurrent deterministic policy gradients[214], and the generator is trained by supervised learning.

This preliminary investigation was unsuccessful insofar that my prototype dynamic scan system does not convincingly outperform static scan systems. However, I believe that it is important to report my progress, despite publication bias against negative results[215–221], as it establishes starting points for further investigation. The main limitation of my scan system is that generator performance is much lower when it is trained for a variety of adaptive scan paths than when it is trained for a single static scan path. For an actor to learn an optimal policy, the generator should ideally be trained until convergence to the highest possible performance for every scan path. However, my generator architecture and learning policy was limited by available computational resources and development time. I also suspect that performance might be improved by replacing RNNs with transformers[222,223] as transformers often achieve similar or higher performance than RNNs[224,225].

There are a variety of additional refinements that could improve training. As an example, RNN computation is delayed by calling a Python function to observe each path segment. Delay could be reduced by more efficient sampling e.g. by using a parallelized routine coded in C/C++; by selecting several possible path segments in advance and selecting the segment that most closely corresponds to an action; or by choosing actions at least one step in advance rather than at each step. In addition, it may help if the generator undergoes additional training iterations in parallel to actor and critic training as improving the generator is critical to improving performance. Finally, increasing generator training iterations may result in overfitting, so it may help to train generators as part of a GAN or introduce other regularization mechanisms. For context, I find that adversarial training can reduce validation divergence[8] (ch. 7) and produce more realistic partial scan completions[5] (ch. 4).

Chapter 6

Improving Electron Micrograph Signal-to-Noise with an Atrous Convolutional Encoder-Decoder

6.1 Scientific Paper

This chapter covers the following paper[7].

> J. M. Ede and R. Beanland. Improving Electron Micrograph Signal-to-Noise with an Atrous Convolutional Encoder-Decoder. *Ultramicroscopy*, 202:18–25, 2019

6.2 Amendments and Corrections

There are amendments or corrections to the paper[7] covered by this chapter.

Location: Page 19, text following eqn 1.
Change: "...to only 25 e^{-2} for a camera..." should say "...to only 25 $e\text{Å}^{-2}$ for a camera...".

Location: Page 21, first paragraph of performance section.
Change: "...structural similarity index (SSIM)..." should say "...structural similarity index measure (SSIM)...".

6.3 Reflection

This chapter covers our paper titled "Improving Electron Micrograph Signal-to-Noise with an Atrous Convolutional Encoder-Decoder"[7] and associated research outputs[16,24,25]. Our paper presents a DNN based on Deeplabv3+ that is trained to remove Poisson noise from TEM images. My DNN is affectionately named "Fluffles" and it is the only DNN that I have named. Pretrained models and performance characterizations are provided for DNNs trained for low and high electron doses. We also show that my DNN has lower MSEs, lower MSE variance, higher SSIMs, and lower or similar SSIM variance to other popular algorithms. We also provide MSE and SSIM distributions, and visualize errors for each output pixel.

Due to limited available computational resources, DNN training was stopped after it surpassed the performance of a variety of popular denoising algorithms. However, there are many other denoising algorithms[226–228] that might achieve higher performance, some of which were developed for electron microscopy[2]. For example, we did not compare our DNN against block-matching and 3D filtering[229,230] (BM3D), which often achieves high-performance. However, an extensive comparison is complicated by source code not being available for some algorithms. In addition, we expect that further training would improve performance as validation errors did not diverge from training errors. For comparison, our DNN was trained for about ten days on two Nvidia GTX 1080 Ti GPUs whereas Xception[231], which is randomly initialized as part of our DNN, was trained for one month on 60 Nvidia K80 GPUs for ImageNet[232] image classification. Indeed, I suspect that restarting DNN training with a pretrained Xception backbone may more quickly achieve much higher performance than continuing training from my pretrained models. Finally, sufficiently deep and wide ANNs are universal approximators[233–241], so denoising DNNs can always outperform or match the accuracy of other methods developed by humans.

A few aspects of my DNN architecture and optimization are peculiar as our paper presents some of my earliest experiments with deep learning. For example, learning rates were stepwise decayed at irregular "wall clock" times. Further, large decreases in errors when learning rates were decreased may indicate that learning rates were too high. Another issue is that ReLU6[242] activation does not significantly outperform ReLU[243,244] activation, so ReLU is preferable as it requires less computation. Finally, I think that my DNN is too large for electron micrograph denoising. We justified that training can be continued and provide pretrained models; however, I doubt that training on the scale of Xception is practical insofar that most electron microscopists do not readily have access to more than a few GPUs for DNN training. I investigated smaller DNNs, which achieved lower performance. However, I expect that their performance could have been improved by further optimization of their training and architecture. In any case, I think that future DNNs for TEM denoising should be developed with automatic machine learning[245–249] (AutoML) as AutoML can balance accuracy and training time, and can often outperform human developers[250,251].

My denoiser has higher errors near output image edges. Higher errors near image edges were also observed for compressed sensing with spiral[5] and uniformly spaced grid[20] scans (ch. 4). Indeed, the structured systematic errors of my denoiser partially motivated my investigations of structured systematic errors in compressed sensing. To avoid higher errors at output edges, I overlap parts of images that my denoiser is applied to so that edges of outputs where errors are higher can be discarded. However, discarding parts of denoiser outputs is computationally inefficient. To reduce structured systematic errors, I tried weighting contributions of output pixel errors to training losses by multiplying pixel errors by their exponential moving averages[5]. However, weighting errors did not have a significant effect. Nevertheless, I expect that higher variation of pixel weights could reduce systematic errors. Moreover, I propose that weights for output pixel errors could be optimized during DNN training to minimize structured systematic errors.

Chapter 7

Exit Wavefunction Reconstruction from Single Transmission Electron Micrographs with Deep Learning

7.1 Scientific Paper

This chapter covers the following paper[8] and its supplementary information[13].

> J. M. Ede, J. J. P. Peters, J. Sloan, and R. Beanland. Exit Wavefunction Reconstruction from Single Transmission Electron Micrographs with Deep Learning. *arXiv preprint arXiv:2001.10938 (under review by Ultramicroscopy)*, 2020
>
> J. M. Ede, J. J. P. Peters, J. Sloan, and R. Beanland. Supplementary Information: Exit Wavefunction Reconstruction from Single Transmission Electron Micrographs with Deep Learning. Zenodo, Online: https://doi.org/10.5281/zenodo.4277357, 2020

7.2 Reflection

This chapter covers our paper titled "Exit Wavefunction Reconstruction from Single Transmission Electron Micrographs with Deep Learning"[8] and associated research outputs[16,26]. At the University of Warwick, EWR is usually based on iterative focal and tilt series reconstruction (FTSR), so a previous PhD student, Mark Dyson, GPU-accelerated FTSR[252]. However, both recording a series of electron micrographs and FTSR usually take several seconds, so FTSR is unsuitable for live EWR. We have an electrostatic biprism that can be used for live in-line holography[253–255]; however, it is not used as we find that in-line holography is more difficult than FTSR. In addition, in-line holography can require expensive microscope modification if a microscope is not already equipped for it. Thus, I was inspired by applications of DNNs to predict missing information for low-light vision[256,257] to investigate live application of DNNs to predict missing phases of exit wavefunctions from single TEM images.

A couple of years ago, it was shown that DNNs can recover phases of exit wavefunctions from single optical micrographs if wavefunctions are constrained by limiting input variety[258–260]. Similarly, electron propagation can be described by wave optics[261], and optical and electron microscopes have similar arrangements of optical and electromagnetic lenses, respectively[262]. Thus, it might be expected that DNNs can recover phases of exit wavefunctions from single TEM images. However, earlier experiments with optical micrographs were unbeknownst to us when we started our investigation. Thus, whether DNNs could reconstruct phase information from single TEM images was contentions as there are infinite possible phases for a given amplitude. Further, previous non-iterative approaches to TEM EWR were limited to defocused images in the Fresnel regime[263] or non-planar incident wavefunctions in the Fraunhofer regime[264].

We were not aware of any large openly accessible datasets containing experimental TEM exit wavefunctions. Consequently, we simulated exit wavefunctions with clTEM[252,265] for a preliminary investigation. Similar to optical EWR[258–260], we found that DNNs can recover the phases of TEM exit wavefunctions if wavefunction variety is restricted. Limitingly, our simulations are unrealistic insofar they do not include aberrations, specimen drift, statistical noise, and higher-order simulation physics. However, we have demonstrated that DNNs can learn to remove noise[7] (ch. 6), specimen drifted can be reduced by sample holders[266], and aberrations can be minimized by aberration correctors[261,267–269]. Moreover, our results present lower bounds for performance as our inputs were far less restricted than possible in practice.

Curating a dataset of experimental exit wavefunctions to train DNNs to recover their phases is time-consuming and expensive. Further, data curation became impractical due to a COVID-19 national lockdown in the United Kingdom[196]. Instead, we propose a new approach to EWR that uses metadata to inform DNN training with single images. Our TEM (ch. 6) and STEM (ch. 4) images in WEMD[3] are provided as a possible resource to investigate our proposal. However, metadata is not included in WEMD, which is problematic as performance is expected to increase with increasing metadata as increasing metadata increasingly restricts probable exit wavefunctions. Nevertheless, DNNs can reconstruct some metadata from unlabelled electron micrographs[270].

Another issue is that experimental WEMD contain images for a range of electron microscope configurations, which would complicate DNN training. For example, experimental TEM images include bright field, dark field, diffraction and CBED images. However, data clustering could be applied to partially automate labelling of electron microscope configurations. For example, I provide pretrained VAEs to embed images for tSNE[3] (ch. 2).

Chapter 8

Conclusions

This thesis covers a subset of my papers on advances in electron microscopy with deep learning. My review paper (ch. 1) offers a substantial introduction that sets my work in context. Ancillary chapters then introduce new machine learning datasets for electron microscopy (ch. 2) and an algorithm to prevent learning instabilty when training large neural networks with limited computational resources (ch. 3). Finally, we report applications of deep learning to compressed sensing in STEM with static (ch. 4) and dynamic (ch. 5) scans, improving TEM signal-to-noise (ch. 6), and TEM exit wavefunction reconstruction (ch. 7). This thesis therefore presents a substantial original contribution to knowledge which is, in practice, worthy of peer-reviewed publication. This thesis adds to my existing papers by presenting their relationships, reflections, and holistic conclusions. To encourage further investigation, source code, pretrained models, datasets, and other research outputs associated with this thesis are openly accessible.

Experiments presented in this thesis are based on unlabelled electron microscopy image data. Thus, this thesis demonstrates that large machine learning datasets can be valuable without needing to add enhancements, such as image-level or pixel-level labels, to data. Indeed, this thesis can be characterized as an investigation into applications of large unlabelled electron microscopy datasets. However, I expect that tSNE clustering based on my pretrained VAE encodings[3] (ch. 2) could ease image-level labelling for future investigations. Most areas of science are facing a reproducibility crisis[115], including artificial intelligence[271], which I think is partly due to a perceived lack of value in archiving data that has not been enhanced. However, this thesis demonstrates that unlabelled data can readily enable new applications of deep learning in electron microscopy. Thus, I hope that my research will encourage more extensive data archiving by the electron microscopy community.

My DNNs were developed with TensorFlow[272,273] and Python. In addition, recent versions of Gatan Microscopy Suite (GMS) software[274], which is often used to drive electron microscopes, support Python[275]. Thus, my pretrained models and source code can be readily integrated into existing GMS software. If a microscope is operated by alternative software or an older version of GMS that does not support Python, TensorFlow supports many other programming languages[2] which can also interface with my pretrained models, and which may be more readily integrated. Alternatively, Python code can often be readily embedded in or executed by other programming languages. To be clear, my DNNs were developed as part of an initial investigation of deep learning in electron microscopy. Thus, this thesis presents lower bounds for performance that may be improved upon by refining ANN architecture and learning policy. Nevertheless, my pretrained models can be the initial basis of deep learning software for electron microscopy.

This thesis includes a variety of experiments to refine ANN architecture and learning policy. As AutoML[245–249] has improved since the start of my PhD, I expect that human involvement can be reduced in future investigations of standard architecture and learning policy variations. However, AutoML is yet to be able to routinely develop new approaches to machine learning, such as VAE encoding normalization and regularization[3] (ch. 2) and ALRC[4] (ch. 3). Most machine learning experts do not think that a technological singularity, where machines outrightly surpasses human developers, is likely for at least a couple of decades[276]. Nonetheless, our increasingly creative machines are already automating some aspects of software development[277,278] and can programmatically describe ANNs[279]. Subsequently, I encourage adoption of creative software, like AutoML, to ease development.

Perhaps the most exciting aspect of ANNs is their scalability[280,281]. Once an ANN has been trained, clones of the ANN and supporting software can be deployed on many electron microscopes at little or no additional cost to the developer. All machine learning software comes with technical debt[282,283]; however, software maintenance costs are usually far lower than the cost of electron microscopes. Thus, machine learning may be a promising means to cheaply enhance electron microscopes. As an example, my experiments indicate that compressed sensing ANNs[5] (ch. 4) can increase STEM and other electron microscopy resolution by up to 10× with minimal information loss. Such a resolution increase could greatly reduce the cost of electron microscopes while maintaining similar capability. Further, I anticipate that multiple ANNs offering a variety of functionality can be combined into a single- or multiple-ANN system that simultaneously offers a variety of enhancements, including increased resolution, decreased noise[7] (ch. 6), and phase information[8] (ch. 6).

I think the main limitation of this thesis, and deep learning, is that it is difficult to fairly compare different approaches to DNN development. As an example, I found that STEM compressed sensing with regularly spaced scans outperforms contiguous scans for the same ANN architecture and learning policy[5] (ch. 4). However, such a performance comparison is complicated by sensitivity of performance to training data, architecture, and learning policy. As a case in point, I argued that contiguous scans could outperform spiral scans if STEM images were not oversampled[5], which could be the case if partial STEM ANNs are also trained to increase image resolution. In part, I think ANN development is an art: Most ANN architecture and learning policy is guided by heuristics, and best approaches to maximize performance are chosen by natural selection[284]. Due to the complicated nature of most data, maximum performances that can be achieved with deep learning are not known. However, it follows from the universal approximator theorem[233–241] that minimum errors can, in principle, be achieved by DNNs.

Applying an ANN to a full image usually requires less computation than applying an ANN to multiple image crops. Processing full images avoids repeated calculations if crops overlap[7] (ch. 6) or lower performance near crop edges where there is less information[5,7,20] (ch. 4 and ch. 6). However, it is usually impractical to train large DNNs to process full electron microscopy images, which are often 1024×1024 or larger, due to limited memory in most GPUs. This was problematic as one of my original agreements about my research was that I would demonstrate that DNNs could be applied to large electron microscopy images, which Richard Beanland and I decided were at least 512×512. As a result, most of my DNNs were developed for 512×512 crops from electron micrographs, especially near the start of my PhD. The combination of large input images and limited available GPU memory restricted training batch sizes to few examples for large ANNs, so I often trained ANNs with a batch size of 1 and either weight[285] or spectral[286] normalization, rather than batch normalization[287].

Most of my DNNs leverage an understanding of physics to add extra information to electron microscopy images. Overt examples include predicting unknown pixels for compressed sensing with static[5] (ch. 4) or adaptive[6] (ch. 5) sparse scans, and unknown phase information from image intensities[8] (ch. 7). More subtly, improving image signal-to-noise with an DNN[7] (ch. 6) is akin to improving signal-to-noise by increasing numbers of intensity measurements. Arguably, even search engines based on VAEs[3] (ch. 2) add information to images insofar that VAE encodings can be compared to quantify semantic similarities between images. Ultimately, my DNNs add information to data that could already be understood from physical laws and observations. However, high-dimensional datasets can be difficult to utilize. Deep learning offers an effective and timely means to both understand high-dimensional data and leverage that understanding to produce results in a useable format. Thus, I both anticipate and encourage further investigation of deep learning in electron microscopy.

References

[1] J. M. Ede. Advances in Electron Microscopy with Deep Learning. *arXiv preprint arXiv:2101.01178*, 2021.

[2] J. M. Ede. Review: Deep Learning in Electron Microscopy. *arXiv preprint arXiv:2009.08328 (accepted by Machine Learning: Science and Technology – https://doi.org/10.1088/2632-2153/abd614)*, 2020.

[3] J. M. Ede. Warwick Electron Microscopy Datasets. *Machine Learning: Science and Technology*, 1(4):045003, 2020.

[4] J. M. Ede and R. Beanland. Adaptive Learning Rate Clipping Stabilizes Learning. *Machine Learning: Science and Technology*, 1:015011, 2020.

[5] J. M. Ede and R. Beanland. Partial Scanning transmission Electron Microscopy with Deep Learning. *Scientific Reports*, 10(1):1–10, 2020.

[6] J. M. Ede. Adaptive Partial Scanning Transmission Electron Microscopy with Reinforcement Learning. *arXiv preprint arXiv:2004.02786 (under review by Machine Learning: Science and Technology)*, 2020.

[7] J. M. Ede and R. Beanland. Improving Electron Micrograph Signal-to-Noise with an Atrous Convolutional Encoder-Decoder. *Ultramicroscopy*, 202:18–25, 2019.

[8] J. M. Ede, J. J. P. Peters, J. Sloan, and R. Beanland. Exit Wavefunction Reconstruction from Single Transmission Electron Micrographs with Deep Learning. *arXiv preprint arXiv:2001.10938 (under review by Ultramicroscopy)*, 2020.

[9] J. M. Ede. Resume of Jeffrey Mark Ede. Zenodo, Online: https://doi.org/10.5281/zenodo.4429077, 2021.

[10] J. M. Ede. Supplementary Information: Warwick Electron Microscopy Datasets. Zenodo, Online: https://doi.org/10.5281/zenodo.3899740, 2020.

[11] J. M. Ede. Supplementary Information: Partial Scanning Transmission Electron Microscopy with Deep Learning. Online: https://static-content.springer.com/esm/

art%3A10.1038%2Fs41598-020-65261-0/MediaObjects/41598_2020_65261_MOESM1_ESM.pdf, 2020.

[12] J. M. Ede. Supplementary Information: Adaptive Partial Scanning Transmission Electron Microscopy with Reinforcement Learning. Zenodo, Online: https://doi.org/10.5281/zenodo.4384708, 2020.

[13] J. M. Ede, J. J. P. Peters, J. Sloan, and R. Beanland. Supplementary Information: Exit Wavefunction Reconstruction from Single Transmission Electron Micrographs with Deep Learning. Zenodo, Online: https://doi.org/10.5281/zenodo.4277357, 2020.

[14] J. M. Ede. Warwick Electron Microscopy Datasets. *arXiv preprint arXiv:2003.01113*, 2020.

[15] J. M. Ede. Source Code for Warwick Electron Microscopy Datasets. Online: https://github.com/Jeffrey-Ede/datasets, 2020.

[16] J. M. Ede. Warwick Electron Microscopy Datasets Archive. Online: https://github.com/Jeffrey-Ede/datasets/wiki, 2020.

[17] J. M. Ede and R. Beanland. Adaptive Learning Rate Clipping Stabilizes Learning. *arXiv preprint arXiv:1906.09060*, 2019.

[18] J. M. Ede. Source Code for Adaptive Learning Rate Clipping Stabilizes Learning. Online: https://github.com/Jeffrey-Ede/ALRC, 2020.

[19] J. M. Ede and R. Beanland. Partial Scanning Transmission Electron Microscopy with Deep Learning. *arXiv preprint arXiv:1910.10467*, 2020.

[20] J. M. Ede. Deep Learning Supersampled Scanning Transmission Electron Microscopy. *arXiv preprint arXiv:1910.10467*, 2019.

[21] J. M. Ede. Source Code for Partial Scanning Transmission Electron Microscopy. Online: https://github.com/Jeffrey-Ede/partial-STEM, 2019.

[22] J. M. Ede. Source Code for Deep Learning Supersampled Scanning Transmission Electron Microscopy. Online: https://github.com/Jeffrey-Ede/DLSS-STEM, 2019.

[23] J. M. Ede. Source Code for Adaptive Partial Scanning Transmission Electron Microscopy with Reinforcement Learning. Online: https://github.com/Jeffrey-Ede/adaptive-scans, 2020.

[24] J. M. Ede. Improving Electron Micrograph Signal-to-Noise with an Atrous Convolutional Encoder-Decoder. *arXiv preprint arXiv:1807.11234*, 2018.

[25] J. M. Ede. Source Code for Improving Electron Micrograph Signal-to-Noise with an Atrous Convolutional Encoder-Decoder. Online: https://github.com/Jeffrey-Ede/Electron-Micrograph-Denoiser, 2019.

[26] J. M. Ede. Source Code for Exit Wavefunction Reconstruction from Single Transmission Electron Micrographs with Deep Learning. Online: https://github.com/Jeffrey-Ede/one-shot, 2019.

[27] J. M. Ede. Progress Reports of Jeffrey Mark Ede: 0.5 Year Progress Report. Zenodo, Online: https://doi.org/10.5281/zenodo.4094750, 2020.

[28] J. M. Ede. Source Code for Beanland Atlas. Online: https://github.com/Jeffrey-Ede/Beanland-Atlas, 2018.

[29] J. M. Ede. Thesis Word Counting. Zenodo, Online: https://doi.org/10.5281/zenodo.4321429, 2020.

[30] J. M. Ede. Posters and Presentations. Zenodo, Online: https://doi.org/10.5281/zenodo.4041574, 2020.

[31] J. M. Ede. Autoencoders, Kernels, and Multilayer Perceptrons for Electron Micrograph Restoration and Compression. *arXiv preprint arXiv:1808.09916*, 2018.

[32] J. M. Ede. Source Code for Autoencoders, Kernels, and Multilayer Perceptrons for Electron Micrograph Restoration and Compression. Online: https://github.com/Jeffrey-Ede/Denoising-Kernels-MLPs-Autoencoders, 2018.

[33] J. M. Ede. Source Code for Simple Webserver. Online: https://github.com/Jeffrey-Ede/simple-webserver, 2019.

[34] Guide to Examinations for Higher Degrees by Research. University of Warwick Doctoral College, Online: https://warwick.ac.uk/services/dc/pgrassessments/gtehdr, 2020.

[35] Regulation 38: Research Degrees. University of Warwick Calendar, Online: https://warwick.ac.uk/services/gov/calendar/section2/regulations/reg38pgr, 2020.

[36] Thesis Writing and Submission. University of Warwick Department of Physics, Online: https://warwick.ac.uk/fac/sci/physics/current/postgraduate/regs/thesis, 2020.

[37] A Warwick Thesis Template. University of Warwick Department of Physics, Online: https://warwick.ac.uk/fac/sci/physics/staff/academic/mhadley/wthesis, 2020.

[38] EPSRC Studentship 1917382: Application of Novel Computing and Data Analysis Methods in Electron Microscopy. UK Research and Innovation, Online: https://gtr.ukri.org/projects?ref=studentship-1917382, 2020.

[39] EPSRC Grant EP/N035437/1: ADEPT – Advanced Devices by ElectroPlaTing. EPSRC, Online: https://gow.epsrc.ukri.org/NGBOViewGrant.aspx?GrantRef=EP/N035437/1, 2016.

[40] A. J. M. Hubert, R. Römer, and R. Beanland. Structure Refinement from 'Digital' Large Angle Convergent Beam Electron Diffraction Patterns. *Ultramicroscopy*, 198:1–9, 2019.

[41] J. M. Ede. Beanland Atlas Repository. Towards Data Science, Online: https://github.com/Jeffrey-Ede/Beanland-Atlas, 2018.

[42] J. L. Hart, S. Liu, A. C. Lang, A. Hubert, A. Zukauskas, C. Canalias, R. Beanland, A. M. Rappe, M. Arredondo, and M. L. Taheri. Electron-Beam-Induced Ferroelectric Domain Behavior in the Transmission Electron Microscope: Toward Deterministic Domain Patterning. *Physical Review B*, 94(17):174104, 2016.

[43] D. Ha. Neural Network Generative Art in Javascript. Online: https://blog.otoro.net/2015/06/19/neural-network-generative-art, 2015.

[44] T. Le. Generate Abstract Random Art with A Neural Network. Medium, Online: https://medium.com/@tuanle618/generate-abstract-random-art-with-a-neural-network-ecef26f3dd5f, 2019.

[45] L. A. Gatys, A. S. Ecker, and M. Bethge. Image Style Transfer Using Convolutional Neural Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2414–2423, 2016.

[46] L. A. Gatys, A. S. Ecker, and M. Bethge. A Neural Algorithm of Artistic Style. *Nature Communications*, 2015.

[47] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. In *European Conference on Computer Vision*, pages 694–711. Springer, 2016.

[48] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro. High-Resolution Image Synthesis and Semantic Manipulation with Conditional GANs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8798–8807, 2018.

[49] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018.

[50] F. Jiang, W. Tao, S. Liu, J. Ren, X. Guo, and D. Zhao. An End-to-End Compression Framework Based on Convolutional Neural Networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(10):3007–3018, 2017.

[51] C. Guerin. Connecting the Dots: Writing a Doctoral Thesis by Publication. In *Research Literacies and Writing Pedagogies for Masters and Doctoral Writers*, pages 31–50. Brill, 2016.

[52] S. Mason, J. E. Morris, and M. K. Merga. Institutional and Supervisory Support for the Thesis by Publication. *Australian Journal of Education*, page 0004944120929065, 2020.

[53] V. Larivière, A. Zuccala, and É. Archambault. The Declining Scientific Impact of Theses: Implications for Electronic Thesis and Dissertation Repositories and Graduate Studies. *Scientometrics*, 74(1):109–121, 2008.

[54] 2018 Global State of Peer Review. Publons, Online: https://publons.com/static/Publons-Global-State-Of-Peer-Review-2018.pdf, 2018.

[55] J. P. Tennant. The State of the Art in Peer Review. *FEMS Microbiology Letters*, 365(19), 2018.

[56] R. Walker and P. Rocha da Silva. Emerging Trends in Peer Review – A Survey. *Frontiers in Neuroscience*, 9:169, 2015.

[57] I. Vesper. Peer Reviewers Unmasked: Largest Global Survey Reveals Trends. *Nature*, 2018.

[58] Z.-Y. Tan, N. Cai, J. Zhou, and S.-G. Zhang. On Performance of Peer Review for Academic Journals: Analysis Based on Distributed Parallel System. *IEEE Access*, 7:19024–19032, 2019.

[59] T. Ferreras-Fernández, F. García-Peñalvo, J. A. Merlo-Vega, and H. Martín-Rodero. Providing Open Access to PhD Theses: Visibility and Citation Benefits. *Program*, 2016.

[60] M. Kettler. Ways of Disseminating, Tracking Usage and Impact of Electronic Theses and Dissertations (ETDs). In *Conference on Grey Literature and Repositories*, page 37, 2016.

[61] B. M. Miller. *The Making of Knowledge-Makers in Composition: A Distant Reading of Dissertations*. PhD thesis, City University of New York, 2015.

[62] University of Warwick Physics PhD Theses. Online: https://wrap.warwick.ac.uk/view/theses/Department_of_Physics.html, 2020.

[63] About arXiv. Online: https://arxiv.org/about, 2020.

[64] P. Ginsparg. ArXiv at 20. *Nature*, 476(7359):145–147, 2011.

[65] G. Pignalberi and M. Dominici. Introduction to LATEX and to Some of its Tools. *ArsTEXnica*, page 8, 2019.

[66] M. Bransen and G. Schulpen. Pimp Your Thesis: A Minimal Introduction to LATEX. IC/TC, U.S.S. Proton, Online: https://ussproton.nl/files/careerweeks/20180320-pimpyourthesis.pdf, 2018.

[67] L. Lamport. *LATEX: A Document Preparation System: User's Guide and Reference Manual*. Addison-Wesley, 1994.

[68] Creative Commons Attribution 4.0 International (CC BY 4.0). Online: https://creativecommons.org/licenses/by/4.0, 2020.

[69] M. B. Hoy. Rise of the Rxivs: How Preprint Servers are Changing the Publishing Process. *Medical Reference Services Quarterly*, 39(1):84–89, 2020.

[70] N. K. Fry, H. Marshall, and T. Mellins-Cohen. In Praise of Preprints. *Microbial Genomics*, 5(4), 2019.

[71] E. G. Rodríguez. Preprints and Preprint Servers as Academic Communication Tools. *Revista Cubana de Información en Ciencias de la Salud*, 30(1), 2019.

[72] G. C. Banks, J. G. Field, F. L. Oswald, E. H. O'Boyle, R. S. Landis, D. E. Rupp, and S. G. Rogelberg. Answers to 18 Questions About Open Science Practices. *Journal of Business and Psychology*, 34(3):257–270, 2019.

[73] N. Fraser, F. Momeni, P. Mayr, and I. Peters. The Relationship Between bioRxiv Preprints, Citations and Altmetrics. *Quantitative Science Studies*, 1(2):618–638, 2020.

[74] Z. Wang, W. Glänzel, and Y. Chen. The Impact of Preprints in Library and Information Science: An Analysis of Citations, Usage and Social Attention Indicators. *Scientometrics*, pages 1–21, 2020.

[75] A. C. Furnival and B. Hubbard. Open Access to Scholarly Communications: Advantages, Policy and Advocacy. *Acceso Abierto a la información en las Bibliotecas Académicas de América Latina y el Caribe*, pages 101–120, 2020.

[76] D. Y. Fu and J. J. Hughey. Meta-Research: Releasing a Preprint is Associated with More Attention and Citations for the Peer-Reviewed Article. *Elife*, 8:e52646, 2019.

[77] Y. Niyazov, C. Vogel, R. Price, B. Lund, D. Judd, A. Akil, M. Mortonson, J. Schwartzman, and M. Shron. Open Access Meets Discoverability: Citations to Articles Posted to Academia.edu. *PLOS ONE*, 11(2):e0148257, 2016.

[78] M. Klein, P. Broadwell, S. E. Farb, and T. Grappone. Comparing Published Scientific Journal Articles to Their Pre-Print Versions. *International Journal on Digital Libraries*, 20 (4):335–350, 2019.

[79] C. F. Carneiro, V. G. Queiroz, T. C. Moulin, C. A. Carvalho, C. B. Haas, D. Rayêe, D. E. Henshall, E. A. De-Souza, F. Espinelli, F. Z. Boos, et al. Comparing Quality of Reporting Between Preprints and Peer-Reviewed Articles in the Biomedical Literature. *BioRxiv*, page 581892, 2019.

[80] Elsevier Language Editing Services. Online: https://webshop.elsevier.com/language-editing-services, 2020.

[81] IOP Editing Services. Online: https://editing.iopscience.iop.org, 2020.

[82] Springer Nature Author Services. Online: https://authorservices.springernature.com, 2020.

[83] Wiley Editing Services. Online: https://wileyeditingservices.com/en, 2020.

[84] R. Roth. Understanding the Importance of Copyediting in Peer-Reviewed Manuscripts. *Science Editor*, 42(2):51, 2019.

[85] ISO 32000-2:2017 Document management — Portable document format — Part 2: PDF 2.0. International Organization for Standardization, Online: https://www.iso.org/standard/51502.html, 2017.

[86] ISO 32000-2:2008 Document management — Portable document format — Part 1: PDF 1.7. Adobe Systems, Online: http://wwwimages.adobe.com/www.adobe.com/content/dam/acom/en/devnet/pdf/pdfs/PDF32000_2008.pdf, 2008.

[87] arXiv License Information. Online: https://arxiv.org/help/license, 2020.

[88] C. B. Clement, M. Bierbaum, K. P. O'Keeffe, and A. A. Alemi. On the Use of ArXiv as a Dataset. *arXiv preprint arXiv:1905.00075*, 2019.

[89] S. Eger, C. Li, F. Netzer, and I. Gurevych. Predicting Research Trends from ArXiv. *arXiv preprint arXiv:1903.02831*, 2019.

[90] T. Ross-Hellauer. What is Open Peer Review? A Systematic Review. *F1000Research*, 6, 2017.

[91] About OpenReview. Online: https://openreview.net/about, 2020.

[92] D. Soergel, A. Saunders, and A. McCallum. Open Scholarship and Peer Review: a Time for Experimentation. In *International Conference on Machine Learning (ICML 2013) Peer Review Workshop*, 2013.

[93] L. Wang and Y. Zhan. A Conceptual Peer Review Model for arXiv and Other Preprint Databases. *Learned Publishing*, 32(3):213–219, 2019.

[94] GitHub Profile of Jeffrey Mark Ede. Online: https://github.com/Jeffrey-Ede, 2020.

[95] Zenodo. Online: https://about.zenodo.org, 2020.

[96] R. Lalli. A Brief History of Physics Reviews. *Nature Reviews Physics*, 1(1):12, 2019.

[97] S. Sengupta, S. Basak, P. Saikia, S. Paul, V. Tsalavoutis, F. Atiah, V. Ravi, and A. Peters. A Review of Deep Learning with Special Emphasis on Architectures, Applications and Recent Trends. *Knowledge-Based Systems*, page 105596, 2020.

[98] A. Shrestha and A. Mahmood. Review of Deep Learning Algorithms and Architectures. *IEEE Access*, 7:53040–53065, 2019.

[99] M. Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M. S. Nasrin, M. Hasan, B. C. Van Essen, A. A. S. Awwal, and V. K. Asari. A State-of-the-Art Survey on Deep Learning Theory and Architectures. *Electronics*, 8(3):292, 2019.

[100] Y. LeCun, Y. Bengio, and G. Hinton. Deep Learning. *Nature*, 521(7553):436–444, 2015.

[101] J. Schmidhuber. Deep Learning in Neural Networks: An Overview. *Neural Networks*, 61: 85–117, 2015.

[102] G. Barbastathis, A. Ozcan, and G. Situ. On the Use of Deep Learning for Computational Imaging. *Optica*, 6(8):921–943, 2019.

[103] M. Ge, F. Su, Z. Zhao, and D. Su. Deep Learning Analysis on Microscopic Imaging in Materials Science. *Materials Today Nano*, page 100087, 2020.

[104] J. Wei, X. Chu, X.-Y. Sun, K. Xu, H.-X. Deng, J. Chen, Z. Wei, and M. Lei. Machine Learning in Materials Science. *InfoMat*, 1(3):338–358, 2019.

[105] G. R. Schleder, A. C. Padilha, C. M. Acosta, M. Costa, and A. Fazzio. From DFT to Machine Learning: Recent Approaches to Materials Science – A Review. *Journal of Physics: Materials*, 2(3):032001, 2019.

[106] G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby, L. Vogt-Maranto, and L. Zdeborová. Machine Learning and the Physical Sciences. *Reviews of Modern Physics*, 91 (4):045002, 2019.

[107] D. P. Kingma and M. Welling. Auto-Encoding Variational Bayes. *arXiv preprint arXiv:1312.6114*, 2014.

[108] D. P. Kingma and M. Welling. An Introduction to Variational Autoencoders. *arXiv preprint arXiv:1906.02691*, 2019.

[109] C. Doersch. Tutorial on Variational Autoencoders. *arXiv preprint arXiv:1606.05908*, 2016.

[110] L. v. d. Maaten and G. Hinton. Visualizing Data Using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.

[111] G. C. Linderman and S. Steinerberger. Clustering with t-SNE, Provably. *SIAM Journal on Mathematics of Data Science*, 1(2):313–332, 2019.

[112] L. van der Maaten. Accelerating t-SNE Using Tree-Based Algorithms. *The Journal of Machine Learning Research*, 15(1):3221–3245, 2014.

[113] L. van der Maaten. Barnes-Hut-SNE. *arXiv preprint arXiv:1301.3342*, 2013.

[114] M. Wattenberg, F. Viégas, and I. Johnson. How to Use t-SNE Effectively. *Distill*, 1(10):e2, 2016.

[115] M. Baker. Reproducibility Crisis. *Nature*, 533(26):353–66, 2016.

[116] C. A. of Cyberspace Studies. Development of World Internet. *World Internet Development Report 2017: Translated by Peng Ping*, pages 1–19, 2019.

[117] T. Berners-Lee, L. Masinter, and M. McCahill. RFC1738: Uniform Resource Locators (URL). *RFC*, 1994. doi: 10.17487/RFC1738.

[118] P. Kaushik, D. P. Singh, and S. Rajpoot. Fibre Optic Communication in 21 st Century. In *2020 International Conference on Intelligent Engineering and Management (ICIEM)*, pages 125–129. IEEE, 2020.

[119] E. Mack. The History of Broadband. *Geographies of the Internet*, pages 63–76, 2020.

[120] L. Abrardi and C. Cambini. Ultra-Fast Broadband Investment and Adoption: A Survey. *Telecommunications Policy*, 43(3):183–198, 2019.

[121] M. Graydon and L. Parks. 'Connecting the Unconnected': A Critical Assessment of US Satellite Internet Services. *Media, Culture & Society*, 42(2):260–276, 2020.

[122] C. Kaufmanna, H.-P. Huthb, F. Zeigerb, and M. Schmidta. Performance Evaluation of Internet over Geostationary Satellite for Industrial Applications. In *70th International Astronautical Congress*. International Astronautical Federation, 2019.

[123] D. Castelvecchi. Google Unveils Search Engine for Open Data. *Nature*, 561(7722):161–163, 2018.

[124] N. Noy. Discovering Millions of Datasets on the Web. The Keyword, Online: `https://blog.google/products/search/discovering-millions-datasets-web`, 2020.

[125] C. Sun, A. Shrivastava, S. Singh, and A. Gupta. Revisiting Unreasonable Effectiveness of Data in Deep Learning Era. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 843–852, 2017.

[126] T. Hey, K. Butler, S. Jackson, and J. Thiyagalingam. Machine Learning and Big Scientific Data. *Philosophical Transactions of the Royal Society A*, 378(2166):20190054, 2020.

[127] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, et al. The Kinetics Human Action Video Dataset. *arXiv preprint arXiv:1705.06950*, 2017.

[128] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[129] S. Abu-El-Haija, N. Kothari, J. Lee, P. Natsev, G. Toderici, B. Varadarajan, and S. Vijayanarasimhan. YouTube-8M: A Large-Scale Video Classification Benchmark. *arXiv preprint arXiv:1609.08675*, 2016.

[130] C. Sohrabi, Z. Alsafi, N. O'Neill, M. Khan, A. Kerwan, A. Al-Jabir, C. Iosifidis, and R. Agha. World Health Organization Declares Global Emergency: A Review of the 2019 Novel Coronavirus (COVID-19). *International Journal of Surgery*, 2020.

[131] W. J. Wiersinga, A. Rhodes, A. C. Cheng, S. J. Peacock, and H. C. Prescott. Pathophysiology, Transmission, Diagnosis, and Treatment of Coronavirus Disease 2019 (COVID-19): A Review. *Jama*, 324(8):782–793, 2020.

[132] T. Singhal. A Review of Coronavirus Disease-2019 (COVID-19). *The Indian Journal of Pediatrics*, pages 1–6, 2020.

[133] L. O. Teixeira, R. M. Pereira, D. Bertolini, L. S. Oliveira, L. Nanni, and Y. M. Costa. Impact of Lung Segmentation on the Diagnosis and Explanation of COVID-19 in Chest X-Ray Images. *arXiv preprint arXiv:2009.09780*, 2020.

[134] A. Ghosal, A. Nandy, A. K. Das, S. Goswami, and M. Panday. A Short Review on Different Clustering Techniques and Their Applications. In *Emerging Technology in Modelling and Graphics*, pages 69–83. Springer, 2020.

[135] M. Z. Rodriguez, C. H. Comin, D. Casanova, O. M. Bruno, D. R. Amancio, L. d. F. Costa, and F. A. Rodrigues. Clustering Algorithms: A Comparative Approach. *PlOS ONE*, 14(1): e0210236, 2019.

[136] K. Djouzi and K. Beghdad-Bey. A Review of Clustering Algorithms for Big Data. In *2019 International Conference on Networking and Advanced Systems (ICNAS)*, pages 1–6. IEEE, 2019.

[137] M. Mittal, L. M. Goyal, D. J. Hemanth, and J. K. Sethi. Clustering Approaches for High-Dimensional Databases: A Review. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 9(3):e1300, 2019.

[138] M. Y. Ansari, A. Ahmad, S. S. Khan, G. Bhushan, et al. Spatiotemporal Clustering: A Review. *Artificial Intelligence Review*, pages 1–43, 2019.

[139] A. K. Jain, M. N. Murty, and P. J. Flynn. Data Clustering: A Review. *ACM Computing Surveys (CSUR)*, 31(3):264–323, 1999.

[140] J. A. Lee and M. Verleysen. *Nonlinear Dimensionality Reduction*. Springer Science & Business Media, 2007.

[141] G. E. Hinton and S. T. Roweis. Stochastic Neighbor Embedding. In *Advances in Neural Information Processing Systems*, pages 857–864, 2003.

[142] W. Li, J. E. Cerise, Y. Yang, and H. Han. Application of t-SNE to Human Genetic Data. *Journal of Bioinformatics and Computational Biology*, 15(04):1750017, 2017.

[143] I. Wallach and R. Lilien. The Protein–Small-Molecule Database, A Non-Redundant Structural Resource for the Analysis of Protein-Ligand Binding. *Bioinformatics*, 25(5):615–620, 2009.

[144] B. M. Devassy and S. George. Dimensionality Reduction and Visualisation of Hyperspectral Ink Data Using t-SNE. *Forensic Science International*, page 110194, 2020.

[145] B. Melit Devassy, S. George, and P. Nussbaum. Unsupervised Clustering of Hyperspectral Paper Data Using t-SNE. *Journal of Imaging*, 6(5):29, 2020.

[146] P. Gang, W. Zhen, W. Zeng, Y. Gordienko, Y. Kochura, O. Alienin, O. Rokovyi, and S. Stirenko. Dimensionality Reduction in Deep Learning for Chest X-Ray Analysis of Lung Cancer. In *2018 Tenth International Conference on Advanced Computational Intelligence (ICACI)*, pages 878–883. IEEE, 2018.

[147] J. Birjandtalab, M. B. Pouyan, and M. Nourani. Nonlinear Dimension Reduction for EEG-Based Epileptic Seizure Detection. In *2016 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*, pages 595–598. IEEE, 2016.

[148] W. M. Abdelmoula, B. Balluff, S. Englert, J. Dijkstra, M. J. Reinders, A. Walch, L. A. McDonnell, and B. P. Lelieveldt. Data-Driven Identification of Prognostic Tumor Subpopulations Using Spatially Mapped t-SNE of Mass Spectrometry Imaging Data. *Proceedings of the National Academy of Sciences*, 113(43):12244–12249, 2016.

[149] F. Psihas, E. Niner, M. Groh, R. Murphy, A. Aurisano, A. Himmel, K. Lang, M. D. Messier, A. Radovic, and A. Sousa. Context-Enriched Identification of Particles with a Convolutional Network for Neutrino Events. *Physical Review D*, 100(7):073005, 2019.

[150] E. Racah, S. Ko, P. Sadowski, W. Bhimji, C. Tull, S.-Y. Oh, P. Baldi, et al. Revealing Fundamental Physics from the Daya Bay Neutrino Experiment Using Deep Neural Networks. In *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 892–897. IEEE, 2016.

[151] F. Gong, F. Bu, Y. Zhang, Y. Yan, R. Hu, and M. Dong. Visual Clustering Analysis of Electricity Data Based on t-SNE. In *2020 IEEE 5th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA)*, pages 234–240. IEEE, 2020.

[152] K. McDonald, M. Tan, and Y. Mann. The Infinite Drum Machine. Experiments with Google, Online: `https://experiments.withgoogle.com/drum-machine`, 2018.

[153] E. Schubert and M. Gertz. Intrinsic t-Stochastic Neighbor Embedding for Visualization and Outlier Detection. In *International Conference on Similarity Search and Applications*, pages 188–203. Springer, 2017.

[154] I. T. Jolliffe and J. Cadima. Principal Component Analysis: A Review and Recent Developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374(2065):20150202, 2016.

[155] M. E. Wall, A. Rechtsteiner, and L. M. Rocha. Singular Value Decomposition and Principal Component Analysis. In *A Practical Approach to Microarray Data Analysis*, pages 91–109. Springer, 2003.

[156] N. Halko, P.-G. Martinsson, and J. A. Tropp. Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions. *SIAM Review*, 53 (2):217–288, 2011.

[157] P.-G. Martinsson, V. Rokhlin, and M. Tygert. A Randomized Algorithm for the Decomposition of Matrices. *Applied and Computational Harmonic Analysis*, 30:47–68, 2011.

[158] N. Pezzotti, J. Thijssen, A. Mordvintsev, T. Höllt, B. Van Lew, B. P. Lelieveldt, E. Eisemann, and A. Vilanova. GPGPU Linear Complexity t-SNE Optimization. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):1172–1181, 2019.

[159] D. M. Chan, R. Rao, F. Huang, and J. F. Canny. t-SNE-CUDA: GPU-Accelerated t-SNE and its Applications to Modern Data. In *2018 30th International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD)*, pages 330–338. IEEE, 2018.

[160] N. Pezzotti, B. P. Lelieveldt, L. van der Maaten, T. Höllt, E. Eisemann, and A. Vilanova. Approximated and User Steerable tSNE for Progressive Visual Analytics. *IEEE Transactions on Visualization and Computer Graphics*, 23(7):1739–1752, 2016.

[161] A. C. Belkina, C. O. Ciccolella, R. Anno, R. Halpert, J. Spidlen, and J. E. Snyder-Cappione. Automated Optimized Parameters for t-Distributed Stochastic Neighbor Embedding Improve Visualization and Analysis of Large Datasets. *Nature Communications*, 10(1):1–12, 2019.

[162] P. Alfeld. A Trivariate Clough—Tocher Scheme for Tetrahedral Data. *Computer Aided Geometric Design*, 1(2):169–181, 1984.

[163] Z. Chen, V. Badrinarayanan, C.-Y. Lee, and A. Rabinovich. GradNorm: Gradient Normalization for Adaptive Loss Balancing in Deep Multitask Networks. *arXiv preprint arXiv:1711.02257*, 2017.

[164] I. Malkiel, S. Ahn, V. Taviani, A. Menini, L. Wolf, and C. J. Hardy. Conditional WGANs with Adaptive Gradient Balancing for Sparse MRI Reconstruction. *arXiv preprint arXiv:1905.00985*, 2019.

[165] W. McIlhagga. Estimates of Edge Detection Filters in Human Vision. *Vision Research*, 153: 30–36, 2018.

[166] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther. Autoencoding Beyond Pixels Using a Learned Similarity Metric. In *International Conference on Machine Learning*, pages 1558–1566, 2016.

[167] G. Grund Pihlgren, F. Sandin, and M. Liwicki. Improving Image Autoencoder Embeddings with Perceptual Loss. In *International Joint Conference on Neural Networks*, 2020.

[168] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004.

[169] H. Zhao, O. Gallo, I. Frosio, and J. Kautz. Loss Functions for Image Restoration with Neural Networks. *IEEE Transactions on Computational Imaging*, 3(1):47–57, 2016.

[170] S. Z. Dadaneh, S. Boluki, M. Yin, M. Zhou, and X. Qian. Pairwise Supervised Hashing with Bernoulli Variational Auto-Encoder and Self-Control Gradient Estimator. *arXiv preprint arXiv:2005.10477*, 2020.

[171] N. Patterson and Y. Wang. Semantic Hashing with Variational Autoencoders, 2016.

[172] G. Jin, Y. Zhang, and K. Lu. Deep Hashing Based on VAE-GAN for Efficient Similarity Retrieval. *Chinese Journal of Electronics*, 28(6):1191–1197, 2019.

[173] F. Mena and R. Ñanculef. A Binary Variational Autoencoder for Hashing. In *Iberoamerican Congress on Pattern Recognition*, pages 131–141. Springer, 2019.

[174] D. Shen, Q. Su, P. Chapfuwa, W. Wang, G. Wang, L. Carin, and R. Henao. Nash: Toward End-to-End Neural Architecture for Generative Semantic Hashing. *arXiv preprint arXiv:1805.05361*, 2018.

[175] S. Chaidaroon and Y. Fang. Variational Deep Semantic Hashing for Text Documents. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 75–84, 2017.

[176] K. J. Liang, C. Li, G. Wang, and L. Carin. Generative Adversarial Network Training is a Continual Learning Problem. *arXiv preprint arXiv:1811.11083*, 2018.

[177] M. Li, M. Soltanolkotabi, and S. Oymak. Gradient Descent with Early Stopping is Provably Robust to Label Noise for Overparameterized Neural Networks. In *International Conference on Artificial Intelligence and Statistics*, pages 4313–4324, 2020.

[178] T. Flynn, K. M. Yu, A. Malik, N. D'Imperio, and S. Yoo. Bounding the Expected Run-Time of Nonconvex Optimization with Early Stopping. *arXiv preprint arXiv:2002.08856*, 2020.

[179] D. Hendrycks, K. Lee, and M. Mazeika. Using Pre-Training can Improve Model Robustness and Uncertainty. *arXiv preprint arXiv:1901.09960*, 2019.

[180] R. Beanland, K. Evans, and R. A. Roemer. Felix. Online: https://github.com/RudoRoemer/Felix, 2020.

[181] G. P. Meyer. An Alternative Probabilistic Interpretation of the Huber Loss. *arXiv preprint arXiv:1911.02088*, 2019.

[182] P. J. Huber. Robust Estimation of a Location Parameter. *The Annals of Mathematical Statistics*, pages 73–101, 1964.

[183] P. Seetharaman, G. Wichern, B. Pardo, and J. L. Roux. AutoClip: Adaptive Gradient Clipping for Source Separation Networks. *arXiv preprint arXiv:2007.14469*, 2020.

[184] R. Pascanu, T. Mikolov, and Y. Bengio. On the Difficulty of Training Recurrent Neural Networks. In *International Conference on Machine Learning*, pages 1310–1318, 2013.

[185] E. Gorbunov, M. Danilova, and A. Gasnikov. Stochastic Optimization with Heavy-Tailed Noise via Accelerated Gradient Clipping. *arXiv preprint arXiv:2005.10785*, 2020.

[186] L. Liu, H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and J. Han. On the Variance of the Adaptive Learning Rate and Beyond. *arXiv preprint arXiv:1908.03265*, 2019.

[187] D. P. Kingma and J. Ba. ADAM: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[188] J. M. Ede. Pixel Subset Super-Compression with a Generative Adversarial Network (Unfinished Manuscript). Zenodo, Online: https://doi.org/10.5281/zenodo.4072946, 2020.

[189] B. Connolly. Atomic Scale Deep Learning. Towards Data Science, Online: https://towardsdatascience.com/atomic-scale-deep-learning-34238feda632, 2020.

[190] J. M. Ede. Pixel Subset Super-Compression of STEM Images. Online: https://zenodo.org/record/4072946#.X37gMWhKiCo, 2020.

[191] X. Sang, A. R. Lupini, R. R. Unocic, M. Chi, A. Y. Borisevich, S. V. Kalinin, E. Endeve, R. K. Archibald, and S. Jesse. Dynamic Scan Control in STEM: Spiral Scans. *Advanced Structural and Chemical Imaging*, 2(1):1–8, 2016.

[192] X. Sang, A. R. Lupini, J. Ding, S. V. Kalinin, S. Jesse, and R. R. Unocic. Precision Controlled Atomic Resolution Scanning Transmission Electron Microscopy Using Spiral Scan Pathways. *Scientific Reports*, 7:43585, 2017.

[193] S. Gandhare and B. Karthikeyan. Survey on FPGA Architecture and Recent Applications. In *2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN)*, pages 1–4. IEEE, 2019.

[194] C. Zhang, B. Berkels, B. Wirth, and P. M. Voyles. Joint Denoising and Distortion Correction for Atomic Column Detection in Scanning Transmission Electron Microscopy Images. *Microscopy and Microanalysis*, 23(S1):164–165, 2017.

[195] P. Jin and X. Li. Correction of Image Drift and Distortion in a Scanning Electron Microscopy. *Journal of Microscopy*, 260(3):268–280, 2015.

[196] E. Aspinall. COVID-19 Timeline. British Foreign Policy Group, Online: https://bfpg.co.uk/2020/04/covid-19-timeline, 2020.

[197] J. Caldeira and B. Nord. Deeply Uncertain: Comparing Methods of Uncertainty Quantification in Deep Learning Algorithms. *arXiv preprint arXiv:2004.10710*, 2020.

[198] A. M. Alaa. Uncertainty Quantification in Deep Learning: Literature Survey. Towards Data Science, Online: https://github.com/ahmedmalaa/deep-learning-uncertainty, 2020.

[199] N. Ståhl, G. Falkman, A. Karlsson, and G. Mathiason. Evaluation of Uncertainty Quantification in Deep Learning. In *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, pages 556–568. Springer, 2020.

[200] A. Loquercio, M. Segu, and D. Scaramuzza. A General Framework for Uncertainty Estimation in Deep Learning. *IEEE Robotics and Automation Letters*, 5(2):3153–3160, 2020.

[201] A. G. Kendall. *Geometry and Uncertainty in Deep Learning for Computer Vision*. PhD thesis, University of Cambridge, 2019.

[202] Y. Gal. *Uncertainty in Deep Learning*. PhD thesis, University of Cambridge, 2016.

[203] C. Rudin. Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and use Interpretable Models Instead. *Nature Machine Intelligence*, 1(5):206–215, 2019.

[204] J. van Amersfoort, L. Smith, Y. W. Teh, and Y. Gal. Simple and Scalable Epistemic Uncertainty Estimation Using a Single Deep Deterministic Neural Network. *arXiv preprint arXiv:2003.02037*, 2020.

[205] B. Lakshminarayanan, A. Pritzel, and C. Blundell. Simple and Scalable Predictive Uncertainty Estimation Using Deep Ensembles. In *Advances in Neural Information Processing Systems*, pages 6402–6413, 2017.

[206] W. J. Maddox, P. Izmailov, T. Garipov, D. P. Vetrov, and A. G. Wilson. A Simple Baseline for Bayesian Uncertainty in Deep Learning. In *Advances in Neural Information Processing Systems*, pages 13153–13164, 2019.

[207] M. Teye, H. Azizpour, and K. Smith. Bayesian Uncertainty Estimation for Batch Normalized Deep Networks. *arXiv preprint arXiv:1802.06455*, 2018.

[208] A. Kendall and Y. Gal. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? In *Advances in Neural Information Processing Systems*, pages 5574–5584, 2017.

[209] Y. Gal and Z. Ghahramani. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In *International Conference on Machine Learning*, pages 1050–1059, 2016.

[210] L. Breiman. Bagging Predictors. *Machine Learning*, 24(2):123–140, 1996.

[211] C. N. d. Santos, Y. Mroueh, I. Padhi, and P. Dognin. Learning Implicit Generative Models by Matching Perceptual Features. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4461–4470, 2019.

[212] N. Saldi, S. Yüksel, and T. Linder. Asymptotic Optimality of Finite Model Approximations for Partially Observed Markov Decision Processes With Discounted Cost. *IEEE Transactions on Automatic Control*, 65(1):130–142, 2019.

[213] T. Jaakkola, S. P. Singh, and M. I. Jordan. Reinforcement Learning Algorithm for Partially Observable Markov Decision Problems. In *Advances in Neural Information Processing Systems*, pages 345–352, 1995.

[214] N. Heess, J. J. Hunt, T. P. Lillicrap, and D. Silver. Memory-Based Control with Recurrent Neural Networks. *arXiv preprint arXiv:1512.04455*, 2015.

[215] B. D. Earp. The Need for Reporting Negative Results - A 90 Year Update. *Journal of clinical and translational research*, 3(Suppl 2):344, 2018.

[216] A. Mlinarić, M. Horvat, and V. Šupak Smolčić. Dealing with the Positive Publication Bias: Why You Should Really Publish Your Negative Results. *Biochemia Medica*, 27(3):447–452, 2017.

[217] S. B. Nissen, T. Magidson, K. Gross, and C. T. Bergstrom. Publication Bias and the Canonization of False Facts. *eLife*, 5:e21451, 2016.

[218] I. Andrews and M. Kasy. Identification of and Correction for Publication Bias. *American Economic Review*, 109(8):2766–94, 2019.

[219] I. Buvat and F. Orlhac. The Dark Side of Radiomics: On the Paramount Importance of Publishing Negative Results. *Journal of Nuclear Medicine*, 60(11):1543–1544, 2019.

[220] H. Sharma and S. Verma. Is Positive Publication Bias Really a Bias, or an Intentionally Created Discrimination Toward Negative Results? *Saudi Journal of Anaesthesia*, 13(4):352, 2019.

[221] N. Matosin, E. Frank, M. Engel, J. S. Lum, and K. A. Newell. Negativity Towards Negative Results: A Discussion of the Disconnect Between Scientific Worth and Scientific Culture, 2014.

[222] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is All You Need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.

[223] J. Alammar. The Illustrated Transformer. GitHub Blog, Online: http://jalammar.github.io/illustrated-transformer, 2018.

[224] S. Karita, N. Chen, T. Hayashi, T. Hori, H. Inaguma, Z. Jiang, M. Someki, N. E. Y. Soplin, R. Yamamoto, X. Wang, et al. A Comparative Study on Transformer vs RNN in Speech Applications. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 449–456. IEEE, 2019.

[225] A. Zeyer, P. Bahar, K. Irie, R. Schlüter, and H. Ney. A Comparison of Transformer and LSTM Encoder Decoder Models for ASR. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 8–15. IEEE, 2019.

[226] B. Goyal, A. Dogra, S. Agrawal, B. Sohi, and A. Sharma. Image Denoising Review: From Classical to State-of-the-Art Approaches. *Information Fusion*, 55:220–244, 2020.

[227] A. Girdher, B. Goyal, A. Dogra, A. Dhindsa, and S. Agrawal. Image Denoising: Issues and Challenges. *Available at SSRN 3446627*, 2019.

[228] L. Fan, F. Zhang, H. Fan, and C. Zhang. Brief Review of Image Denoising Techniques. *Visual Computing for Industry, Biomedicine, and Art*, 2(1):7, 2019.

[229] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering. *IEEE Transactions on image processing*, 16(8): 2080–2095, 2007.

[230] M. Lebrun. An Analysis and Implementation of the BM3D Image Denoising Method. *Image Processing On Line*, 2:175–213, 2012.

[231] F. Chollet. Xception: Deep Learning with Depthwise Separable Convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1251–1258, 2017.

[232] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012.

[233] P. Kidger and T. Lyons. Universal Approximation with Deep Narrow Networks. *arXiv preprint arXiv:1905.08539*, 2019.

[234] H. Lin and S. Jegelka. ResNet with One-Neuron Hidden Layers is a Universal Approximator. In *Advances in Neural Information Processing Systems*, pages 6169–6178, 2018.

[235] B. Hanin and M. Sellke. Approximating Continuous Functions by ReLU Nets of Minimal Width. *arXiv preprint arXiv:1710.11278*, 2017.

[236] Z. Lu, H. Pu, F. Wang, Z. Hu, and L. Wang. The Expressive Power of Neural Networks: A View from the Width. In *Advances in Neural Information Processing Systems*, pages 6231–6239, 2017.

[237] A. Pinkus. Approximation Theory of the MLP Model in Neural Networks. *Acta Numerica*, 8(1):143–195, 1999.

[238] M. Leshno, V. Y. Lin, A. Pinkus, and S. Schocken. Multilayer Feedforward Networks with a Nonpolynomial Activation Function can Approximate any Function. *Neural Networks*, 6(6): 861–867, 1993.

[239] K. Hornik. Approximation Capabilities of Multilayer Feedforward Networks. *Neural Networks*, 4(2):251–257, 1991.

[240] K. Hornik, M. Stinchcombe, and H. White. Multilayer Feedforward Networks are Universal Approximators. *Neural Networks*, 2(5):359–366, 1989.

[241] G. Cybenko. Approximation by Superpositions of a Sigmoidal Function. *Mathematics of Control, Signals and Systems*, 2(4):303–314, 1989.

[242] A. Krizhevsky. Convolutional Deep Belief Networks on CIFAR-10. *Technical Report*, 40(7): 1–9, 2010.

[243] V. Nair and G. E. Hinton. Rectified Linear Units Improve Restricted Boltzmann Machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 807–814, 2010.

[244] X. Glorot, A. Bordes, and Y. Bengio. Deep Sparse Rectifier Neural Networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 315–323, 2011.

[245] X. He, K. Zhao, and X. Chu. AutoML: A Survey of the State-of-the-Art. *arXiv preprint arXiv:1908.00709*, 2019.

[246] E. Malekhosseini, M. Hajabdollahi, N. Karimi, and S. Samavi. Modeling Neural Architecture Search Methods for Deep Networks. *arXiv preprint arXiv:1912.13183*, 2019.

[247] Y. Jaafra, J. L. Laurent, A. Deruyver, and M. S. Naceur. Reinforcement Learning for Neural Architecture Search: A Review. *Image and Vision Computing*, 89:57–66, 2019.

[248] T. Elsken, J. H. Metzen, and F. Hutter. Neural Architecture Search: A Survey. *arXiv preprint arXiv:1808.05377*, 2018.

[249] J. Waring, C. Lindvall, and R. Umeton. Automated Machine Learning: Review of the State-of-the-Art and Opportunities for Healthcare. *Artificial Intelligence in Medicine*, page 101822, 2020.

[250] M. Hanussek, M. Blohm, and M. Kintz. Can AutoML Outperform Humans? An Evaluation on Popular OpenML Datasets Using AutoML Benchmark. *arXiv preprint arXiv:2009.01564*, 2020.

[251] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le. Learning Transferable Architectures for Scalable Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8697–8710, 2018.

[252] M. A. Dyson. *Advances in Computational Methods for Transmission Electron Microscopy Simulation and Image Processing*. PhD thesis, University of Warwick, 2014.

[253] M. Lehmann and H. Lichte. Tutorial on Off-Axis Electron Holography. *Microscopy and Microanalysis*, 8(6):447–466, 2002.

[254] C. T. Koch and A. Lubk. Off-Axis and Inline Electron Holography: A Quantitative Comparison. *Ultramicroscopy*, 110(5):460–471, 2010.

[255] C. Ozsoy-Keskinbora, C. B. Boothroyd, R. Dunin-Borkowski, P. A. Van Aken, and C. T. Koch. Hybridization Approach to In-Line and Off-Axis (Electron) Holography for Superior Resolution and Phase Sensitivity. *Scientific Reports*, 4:7020, 2014.

[256] F. Almasri and O. Debeir. Robust Perceptual Night Vision in Thermal Colorization. *arXiv preprint arXiv:2003.02204*, 2020.

[257] C. Chen, Q. Chen, J. Xu, and V. Koltun. Learning to See in the Dark. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3291–3300, 2018.

[258] Y. Rivenson, Y. Zhang, H. Günaydın, D. Teng, and A. Ozcan. Phase Recovery and Holographic Image Reconstruction Using Deep Learning in Neural Networks. *Light: Science & Applications*, 7(2):17141–17141, 2018.

[259] Y. Wu, Y. Rivenson, Y. Zhang, Z. Wei, H. Günaydin, X. Lin, and A. Ozcan. Extended Depth-of-Field in Holographic Imaging Using Deep-Learning-Based AutofocUsing and Phase Recovery. *Optica*, 5(6):704–710, 2018.

[260] A. Sinha, J. Lee, S. Li, and G. Barbastathis. Lensless Computational Imaging Through Deep Learning. *Optica*, 4(9):1117–1125, 2017.

[261] H. H. Rose. Optics of High-Performance Electron Microscopes. *Science and Technology of Advanced Materials*, 9(1):014107, 2008.

[262] X. Chen, B. Zheng, and H. Liu. Optical and Digital Microscopic Imaging Techniques and Applications in Pathology. *Analytical Cellular Pathology*, 34(1, 2):5–18, 2011.

[263] A. J. Morgan, A. V. Martin, A. J. D'Alfonso, C. T. Putkunz, and L. J. Allen. Direct Exit-Wave Reconstruction from a Single Defocused Image. *Ultramicroscopy*, 111(9-10):1455–1460, 2011.

[264] A. V. Martin and L. J. Allen. Direct Retrieval of a Complex Wave from its Diffraction Pattern. *Optics Communications*, 281(20):5114–5121, 2008.

[265] J. J. P. Peters and M. A. Dyson. clTEM. Online: https://github.com/JJPPeters/clTEM, 2019.

[266] B. H. Goodge, E. Bianco, and H. W. Kourkoutis. Atomic-Resolution Cryo-STEM Across Continuously Variable Temperature. *arXiv preprint arXiv:2001.11581*, 2020.

[267] S. J. Pennycook. The Impact of STEM Aberration Correction on Materials Science. *Ultramicroscopy*, 180:22–33, 2017.

[268] Q. M. Ramasse. Twenty Years After: How "Aberration Correction in the STEM" Truly Placed a "A Synchrotron in a Microscope". *Ultramicroscopy*, 180:41–51, 2017.

[269] P. Hawkes. Aberration Correction Past and Present. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 367(1903):3637–3664, 2009.

[270] G. H. Weber, C. Ophus, and L. Ramakrishnan. Automated Labeling of Electron Microscopy Images Using Deep Learning. In *Proceedings of MLHPC 2018: Machine Learning in HPC Environments, Held in conjunction with SC 2018: The International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 26–36. IEEE, 2018.

[271] M. Hutson. Artificial Intelligence Faces Reproducibility Crisis. *Science*, 359(6377):725–726, 2018.

[272] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. TensorFlow: A System for Large-Scale Machine Learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pages 265–283, 2016.

[273] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. *arXiv preprint arXiv:1603.04467*, 2016.

[274] Gatan Microscopy Suite Software. Online: www.gatan.com/products/tem-analysis/gatan-microscopy-suite-software, 2020.

[275] B. Miller and S. Mick. Real-Time Data Processing Using Python in DigitalMicrograph. *Microscopy and Microanalysis*, 25(S2):234–235, 2019.

[276] V. C. Müller and N. Bostrom. Future Progress in Artificial Intelligence: A Survey of Expert Opinion. In *Fundamental Issues of Artificial Intelligence*, pages 555–572. Springer, 2016.

[277] A. Sarkar and S. Cooper. Towards Game Design via Creative Machine Learning (GDCML). In *2020 IEEE Conference on Games (CoG)*, pages 744–751. IEEE, 2020.

[278] M. Guzdial, N. Liao, and M. Riedl. Co-Creative Level Design via Machine Learning. *arXiv preprint arXiv:1809.09420*, 2018.

[279] A. Sethi, A. Sankaran, N. Panwar, S. Khare, and S. Mani. DLPaper2Code: Auto-Generation of Code from Deep Learning Research Papers. *arXiv preprint arXiv:1711.03543*, 2017.

[280] L. E. Lwakatare, A. Raj, I. Crnkovic, J. Bosch, and H. H. Olsson. Large-Scale Machine Learning Systems in Real-World Industrial Settings: A Review of Challenges and Solutions. *Information and Software Technology*, 127:106368, 2020.

[281] P. Gupta, A. Sharma, and R. Jindal. Scalable Machine-Learning Algorithms for Big Data Analytics: A Comprehensive Review. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 6(6):194–214, 2016.

[282] E. Breck, S. Cai, E. Nielsen, M. Salib, and D. Sculley. The ML Test Score: A Rubric for ML Production Readiness and Technical Debt Reduction. In *2017 IEEE International Conference on Big Data (Big Data)*, pages 1123–1132. IEEE, 2017.

[283] D. Sculley, G. Holt, D. Golovin, E. Davydov, T. Phillips, D. Ebner, V. Chaudhary, M. Young, J.-F. Crespo, and D. Dennison. Hidden Technical Debt in Machine Learning Systems. In *Advances in Neural Information Processing Systems*, pages 2503–2511, 2015.

[284] B. R. Johnson and S. K. Lam. Self-Organization, Natural Selection, and Evolution: Cellular Hardware and Genetic Software. *Bioscience*, 60(11):879–885, 2010.

[285] T. Salimans and D. P. Kingma. Weight Normalization: A Simple Reparameterization to Accelerate Training of Deep Neural Networks. In *Advances in Neural Information Processing Systems*, pages 901–909, 2016.

[286] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral Normalization for Generative Adversarial Networks. *arXiv preprint arXiv:1802.05957*, 2018.

[287] S. Ioffe and C. Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv preprint arXiv:1502.03167*, 2015.