

UC Santa Barbara

UC Santa Barbara Electronic Theses and Dissertations

Title

Enhancing Materials Microstructure Analysis with Physics-Informed Computer Vision

Permalink

<https://escholarship.org/uc/item/9kd7v583>

Author

Jangid, Devendra Kumar

Publication Date

2024

Peer reviewed|Thesis/dissertation

University of California
Santa Barbara

Enhancing Materials Microstructure Analysis with Physics-Informed Computer Vision

A dissertation submitted in partial satisfaction
of the requirements for the degree

Doctor of Philosophy

in

Electrical and Computer Engineering

by

Devendra Kumar Jangid

Committee in charge:

Professor B.S. Manjunath, Chair
Professor Kenneth Rose
Professor Shiv Chandrasekaran
Professor Samantha Daly
Professor Tresa Pollock

March 2024

The Dissertation of Devendra Kumar Jangid is approved.

Professor Kenneth Rose

Professor Shiv Chandrasekaran

Professor Samantha Daly

Professor Tresa Pollock

Professor B.S. Manjunath, Committee Chair

March, 2024

Enhancing Materials Microstructure Analysis with Physics-Informed Computer Vision

Copyright © 2024

by

Devendra Kumar Jangid

Dedicated to my family and friends.

Acknowledgements

I would like to take a moment to express my heartfelt gratitude to all those who have supported me throughout my dissertation journey.

First and foremost, my deepest appreciation goes to Professor B.S. Manjunath, whose invaluable guidance, patience, and inspiration have been pivotal in my research. His mentorship has not only helped me to carve out my niche in the fields of computer vision and machine learning but has also taught me the intricacies of conducting impactful research.

I am also immensely grateful to my dissertation committee members, Professor Tresa Pollock, Professor Samantha Daly, Professor Kenneth Rose, and Professor Shiv Chandrasekaran. Their insightful discussions and valuable suggestions have significantly enriched my research experience. I extend special thanks to Professor Samantha Daly and Professor Tresa Pollock for creating an excellent environment conducive to interdisciplinary research. Additionally, I am thankful for the financial support from NSF SI2-SSI award number 1664172 and NSF award number 1934641, which have been instrumental in my Ph.D. studies.

Collaborating with Dr. Neal Brodnik and Dr. McLean Echlin on interdisciplinary projects has been a rewarding experience. My sincere thanks to Dr. Neal Brodnik, Dr. McLean Echlin for their contributions on Material Science to our collaborative work. I am also grateful to the members of the VRL lab – Michael Goebel, Amil Khan, Joaquin Giorgi, Po-Yu Kao, Satish Kumar, Raphael Santos, Sai Siddharth, Connor Levenson, Chandrakanth Gudavalli, Shailja, Abu Saleh Mohammed Iftexhar Niloy, Bowen Zhang, Ivan Arevalo, Oytun Ulutan, Austin Mcever, Angela Zhang, Ekta Prashnani – for their unwavering support and camaraderie. I would like to acknowledge Patrick Callahan, Toby Francis, Andrew Polonsky, and Joseph Wendorf for their efforts in collecting the

3D Ti-6Al-4V and Ti-7Al datasets, which have been crucial to my research.

Lastly, but most importantly, my heartfelt thanks go to my family, especially my parents, whose love and unwavering support have been my constant source of strength. I could not have reached this far without their unconditional support. I am also profoundly grateful to all my friends who have stood by me throughout this journey. Your support has been invaluable.

Curriculum Vitæ

Devendra Kumar Jangid

Education

March 2024	Doctor of Philosophy Electrical and Computer Engineering University of California, Santa Barbara, USA.
June 2020	Master of Science Electrical and Computer Engineering University of California, Santa Barbara, USA.
June 2015	Bachelor of Technology Electrical Engineering Indian Institute of Technology, Jodhpur.

Honors & Awards

2023	Best Paper Award at International Conference on Machine Learning Workshop
2023	Spotlight Presentation at International Conference on Machine Learning Workshop
2023	ECE Department Dissertation Fellowship, UCSB
2023	UCSB Doctoral Travel Grant for ICML 2023
2023	Conference Travel Grant, UCSB Graduate Student Association
2019	Outstanding Teaching Assistant Award from ECE department of UC, Santa Barbara
2018	ECE Department Fellowship for Fall 2018
2014	DAAD-WISE Scholarship from the German Federal government for Summer Intern
2011-2015	Merit Cum Mean Scholarship for four years in B.Tech from IIT Jodhpur

Publications and Patents

D.K.Jangid, N.R.Brodnik, M.P. Echlin, T.M. Pollock, S. Daly, B.S. Manjunath "Q-RBSA: High-Resolution 3D EBSD Maps Generation using Efficient Quaternion Transformer Network", Nature Partner Journal, Computational Materials (2024)

D.K.Jangid, N.R.Brodnik, M.P. Echlin, T.M. Pollock, S. Daly, B.S. Manjunath "Titanium 3D Microstructure for Physics-based Generative Models: A Dataset and Primer", Syms and ML Workshop, International Conference on Machine Learning (2023)

D.K.Jangid, N.R.Brodnik, M.G. Goebel, A.Khan, S.S. Majeti, M.P. Echlin, T.M. Pollock, S. Daly, B.S. Manjunath "Adaptable Physics-Based Super-Resolution for Electron Backscatter Diffraction Maps", Nature Partner Journal, Computational Materials 8, 255 (2022)

D.K.Jangid, N.R.Brodnik, A.Khan, M.P. Echlin, T.M. Pollock, S. Daly, B.S. Manjunath "3D Grain Shape Generation in Polycrystals using Generative Adversarial Networks ", Integrating Materials and Manufacturing Innovation Journal 11, 71-84 (2022)

D.K.Jangid, A Gnanasambandam, JW Glotzbach, JS Lee, HR Sheikh, "Efficient Single Image Super-resolution for Smartphone Camera" (2023) (filed to US Patent Office)

D.K. Jangid, JS Lee, HR Sheikh, "Machine learning-based approaches for synthetic training data generation and image sharpening" (2022) (filed to US Patent Office)

Experience

01/2020 - 03/2024	Graduate Student Researcher, Vision Research Lab, UCSB
09/2018 - 12/2019	Teaching Assistant, Electrical and Computer Engineering, UCSB
08/2015 - 08/2018	Scientist, Defence Research and Development Organization, India
06/2022 - 09/2022	Research Intern, Samsung Research America
06/2021 - 09/2021	Research Intern, Samsung Research America
05/2014 - 7/2014	Research Intern, Jacobs University, Bremen, Germany

Technical Skills

Python, Matlab, C++, C, Bash, PyTorch, TensorFlow, Numpy, Git, Linux, Docker

Service

2023	Reviewer of Synergy of Scientific Modeling and Machine Learning workshop at ICML Conference 2023, AI4Mat workshop at NeurIPS Conference 2023
------	--

Abstract

Enhancing Materials Microstructure Analysis with Physics-Informed Computer Vision

by

Devendra Kumar Jangid

Collecting 3D microstructural information of materials poses significant challenges, being a process that is time-consuming and expensive. While advancements in serial sectioning instrumentation have expedited the acquisition of 3D microstructure data, the process of obtaining crystallographic information through electron backscatter diffraction (EBSD) imaging continues to be a bottleneck, limiting the overall rate of data collection. In this research, we explore physics-informed computer vision methods to generate high-resolution 3D microstructure data. These methods are a primary step in solving the cost and time challenges. EBSD is a scanning electron microscope (SEM) imaging modality that maps crystal lattice orientation by analyzing diffraction patterns. EBSD maps are used to determine the microstructural properties such as texture, orientation gradients, phase distributions, and point-to-point orientation correlations, all of which are critical for understanding material performance. EBSD maps contain information about crystal 3d orientation in Euler space that follows crystallography symmetry properties. However, it is difficult to compute rotational distance and symmetry of 3d crystal orientation in Euler space. To solve these unique computational challenges associated with rotational distance and symmetry, we developed a physics-inspired 3D deep learning framework that uses rotational symmetry and quaternion orientation space as priors to generate high-resolution microstructure. The proposed quaternion residual block self-attention neural network (QRBSA) with physics-guided crystal symmetry loss is used to super-resolve high-resolution 3D microstructure from sparsely sectioned EBSD maps. We demonstrate,

both qualitatively and quantitatively, that integrating the physics of microstructure into the deep learning architecture and loss function significantly reduces superresolution synthesis error compared to standard deep learning networks and loss functions.

Additionally, we propose a 3D Generative Adversarial Network (GAN) framework known as M(Material)-GAN, which can be used to learn the morphologies of 3D grains and synthesize realistic grains in microstructures. The moment invariances are used to quantitatively evaluate the generated grains and real grains. The creation of synthetic 3D grains represents a foundational step towards generating comprehensive synthetic 3D microstructures through deep learning techniques. The data and methods developed are available to the broader research community through the UCSB BisQue platform.

Contents

Curriculum Vitae	vii
Abstract	ix
List of Figures	xiii
List of Tables	xx
1 Material Microstructure in Process-Structure-Property	1
1.1 3D Microstructure Data	2
1.2 High-Resolution 3D Microstructure Data Generation	3
1.3 Single Grain Generation for Synthetic 3D Microstructure Generation	5
1.4 Summary of Contributions	6
1.5 Thesis Outline	7
2 Microstructure Datasets	10
2.1 Microstructure Dataset	11
2.2 Microstructure Dataset Representation	14
2.3 Titanium Microstructure	21
2.4 Accessing Microstructure Dataset on BisQue	22
2.5 Dream 3D Pipeline	26
2.6 Conclusion	26
3 Adaptable Physics-based Super-resolution for Electron Backscatter Diffraction Maps	27
3.1 Introduction	28
3.2 Method	32
3.3 Results	44
3.4 EBSD SR BisQue Module	55
3.5 Discussion and Conclusion	56

4	Q-RBSA: High-Resolution 3D EBSD Map Generation Using An Efficient Quaternion Transformer Network	59
4.1	Introduction	60
4.2	Method	63
4.3	Results	77
4.4	Q-RBSA BisQue Module	85
4.5	Discussion and Conclusion	86
4.6	Appendix	90
5	3D Grain Shape Generation in Polycrystals Using Generative Adversarial Networks	95
5.1	Introduction	96
5.2	Previous Studies in 3D Object Generation	100
5.3	Method	105
5.4	Results	115
5.5	M-GAN BisQue Module	120
5.6	Discussion	120
5.7	Conclusions	124
5.8	Appendix	124
6	Conclusions and Future Work	130
6.1	Conclusions	130
6.2	Future Work	132
	Bibliography	136

List of Figures

2.1	TRIBEAM MICROSCOPE: 3D microstructure datasets are obtained during a serial sectioning experiment. The TriBeam is so named due to the presence of three beams: the electron, focused ion, and laser. The electron beam is used to gather diffraction patterns, whereas the laser beam is used to ablate the materials, layer-by-layer. The ion beam can be used clean the laser ablated surface in damage sensitive materials, if necessary. . . .	12
2.2	MICROSTRUCTURE DATA COLLECTION PROCESS: Material researchers collect Kikuchi diffraction patterns for each (x, y, z) coordinate of materials with the given step sizes (s_x, s_y, s_z) in (x, y, z) axis. The Kikuchi diffraction patterns are then indexed to determine the crystal orientation at each point. This information about crystal orientation is useful for predicting material properties. The inverse Pole Figure (IPF) technique is used to visualize the 3D crystal orientation.	13
2.3	INVERSE POLE FIGURE (IPF) MAP OF TI-6AL-4V-EQUIAXED MICROSTRUCTURE DATASET: The color represents the orientation of crystal. Each grain of the same color has a nearly identical crystallographic orientation. Rolling direction (RD), transverse direction (TD) and normal direction (ND) reference arrows are also shown. IPF coloring is referenced to the TD.	19
2.4	RENDERING OF 3D EBSD DATASET INVESTIGATED: shown in IPF coloring of the titanium alloys, (a) Ti-6Al-4V and (b) Ti-7Al mechanically loaded in tension to 1% strain and (c) 3%, used for network training, testing and validation. Dataset details are available elsewhere [1, 2]. . . .	21
2.5	TI-6AL-4V MICROSTRUCTURE DATASET ON BISQUE: Ti-6Al-4V 3D microstructure datasets can be accessed on our open source web-based architecture "BisQue". The microstructure datasets are saved as hdf/dream3d format and can visualized using "IPFColor" field. The crystal orientations at each voxel is saved in "EulerAngles" and "Quaternions" fields. "FeatureIds" field shows the location of grains, and each grain has a distinct feature id.	23
2.6	FILE STRUCTURE OF TI-6AL-4V MICROSTRUCTURE DATASET ON BISQUE	24

3.1	Training Set	35
3.2	Validation Set	35
3.3	Test Set	35
3.4	PRE-PROCESSING OF Ti-6Al-4V DATASETS. Figure 3.1 has size of $300 \times 124 \times 408 \times 4$. Figure 3.2 and Figure 3.3 are divided into 3 mutually orthogonal parallelepiped blocks, which are then sliced into images.	35
3.5	RENDERING OF 3D EBSD DATASET INVESTIGATED: shown in IPF coloring of the titanium alloys, (a) Ti-6Al-4V and (b) Ti-7Al mechanically loaded in tension to 1% strain and (c) 3%, used for network training, testing and validation. Dataset details are available elsewhere [1, 2].	35
3.6	NETWORK ARCHITECTURE: Training Pipeline: A low-resolution EBSD map in quaternion orientation space is given to an image super-resolution network architecture that generates a high-resolution EBSD map in quaternion orientation space. A crystallographic symmetry physics-based loss with L_1 or approximate rotational distance is used during training. Inference Pipeline: The image super-resolution network generates a high-resolution EBSD map in quaternion orientation space, which is reduced to a fundamental zone space, and converted to Euler orientation space to visualize in IPF color map. Symmetry Loss: Takes all possible hexagonal symmetries for the titanium alloy, and computes the minimum distance between all possible generated output and ground truths. The distance can be L_1 or approximate rotational distance.	36
3.7	ORIENTATION LOSS DISTRIBUTIONS: Probability distribution of loss distances between pairs of randomly sampled 3D rotation vectors. L_1 distance is shown in blue, L_1 distance accounting for crystal symmetry is shown in green, and approximate rotational distance accounting for crystal symmetry is shown in red.	39
3.8	APPROXIMATION OF THE ROTATIONAL DISTANCE EQUATION: The derivative is not defined at $d_{\text{euclid}} = 2$, so a linear approximation is computed to ensure smooth loss behavior.	40
3.9	NETWORK OUTPUT COMPARISON WITH DIFFERENT PATCH SIZES: Larger patch sizes lead to improvement across all architectures.	43
3.10	NETWORK OUTPUT COMPARISON WITH DIFFERENT BATCH SIZES: Larger batch size leads to lower quality results.	43
3.11	COMPARISON OF HR GROUND TRUTH TO TRADITIONAL UPSCALING: Bilinear, bicubic, and nearest neighbor upscaling produce inferior results. Bilinear and bicubic results are nonphysical, and nearest neighbor results are visually identical to LR input.	44
3.12	VISUAL COMPARISON OF 4X EBSD-SR ON A TEST EBSD MAP: L_1 has non physical structure at grain boundaries, and L_1 with symmetry and approximate rotational with symmetry reduce the non-physical structures at grain boundaries.	44

3.13	VISUAL COMPARISON OF 4X EBSD-SR ON A TEST EBSD MAP: L_1 has non physical structure at grain boundaries, and L_1 with symmetry and approximate rotational with symmetry reduce the non-physical structures at grain boundaries.	48
3.14	VISUAL COMPARISON OF 4X EBSD-SR ON A TEST EBSD MAP: L_1 has non physical structure at grain boundaries, and L_1 with symmetry and approximate rotational with symmetry reduce the non-physical structures at grain boundaries.	49
3.15	COMPARISON OF UPSAMPLING LAYERS FOR TI-6AL-4V DATASET: Comparison of output of network trained on Ti-6Al-4V with different upsampling layers in upsampling and reconstruction part of network architecture	52
3.16	COMPARISON OF UPSAMPLING LAYERS FOR TI-7AL 1 % DATASET: Comparison of output of network trained on Ti-7Al 1 % with different upsampling layers in upsampling and reconstruction part of network architecture	53
3.17	COMPARISON OF UPSAMPLING LAYERS FOR TI-7AL 3 % DATASET: Comparison of output of network trained on Ti-7Al 3 % with different upsampling layers in upsampling and reconstruction part of network architecture	54
3.18	BISQUE MODULE FOR EBSD SR: The input for the BisQue model is a low-resolution Electron Backscatter Diffraction (EBSD) map in numpy format. This module efficiently processes the input to generate a high-resolution EBSD map, also in numpy format. Additionally, it facilitates the visualization of the EBSD map through an Inverse Pole Figure (IPF) map, offering a clear and detailed representation.	56
4.1	Q-RBSA EBSD RESOLUTION ENHANCEMENT FRAMEWORK: In the experimental pipeline shown in (a), material researchers collect EBSD orientation information for each (x,y) coordinate in a given sectioning plane, and then remove material using laser ablation or robotic polishing to reach the next plane in the z direction to build a 3D volume. In our framework (b), researchers collect EBSD information from a reduced set of points (blue planes), omitting some planes that would normally be gathered (gray planes). The missing information (green planes) are then generated in 2D as a series of (x,z) or (y,z) planes by our quaternion-based, physics-informed deep learning framework, shown in (c). Here, the network takes advantage of orthogonal independence to efficiently generate 3D volumes using less data, as large amounts of EBSD are costly and the choice of serial sectioning direction has minimal impact on the resultant final volume.	64

4.2	RENDERING OF 3D EBSD DATASET INVESTIGATED: shown in IPF coloring of the titanium alloys, (a) Ti-6Al-4V and (b) Ti-7Al mechanically loaded in tension to 1% strain and (c) 3%, used for network training, testing and validation. Dataset details are available elsewhere [1,2]. . . .	66
4.3	QUATERNION RESIDUAL BLOCK SELF-ATTENTION (QRBSA) NETWORK: A sparsely sectioned 2D EBSD map is given to the QRBSA network (a) to generate a high-resolution 2D EBSD map. QRBSA consists of three parts: a Shallow feature extractor, a Deep feature extractor, and Upsampling and Reconstruction. The deep feature extractor uses a residual architecture (b) where residual self-attention blocks (c) are modified with quaternion convolution layers and transformer blocks (d) to efficiently handle orientation data. Quaternion convolution is used to learn local-level relationships, while quaternion transformer blocks learn the global statistics of feature maps. Pixelshuffle layer, modified for 1-dimensional upsampling, is used in the upsampling and reconstruction block to upsample feature maps. .	70
4.4	QUATERNION SELF ATTENTION: Self-Attention in (a) is computed using quaternion convolution across feature dimension instead of spatial dimension to reduce computational complexity to linear. A transposed attention map (A) is calculated from reshaped query (Q_r) and reshaped key (K_r). A quaternion self-attention is computed from the transposed attention map (A) and reshaped value (V_r). QUATERNION FEED FORWARD NETWORK: Shown in (b), performs controlled feature transformation to allow useful information to propagate further using gated quaternion convolution. . .	73
4.5	VISUAL COMPARISON OF NETWORK OUTPUT FOR EXAMPLE 2D EBSD MAPS WITH A SCALE FACTOR OF 4: The predicted EBSD maps (Network Output) from the QRBSA network are similar to the ground truth EBSD maps in for both the Ti-6Al-4V dataset (a) and both Ti-7Al datasets (b) and (c). The black rows correspond to the missing data in the sparsely sectioned input EBSD maps. In this case, one row of EBSD data is used for every three rows of missing EBSD data.	77
4.6	NEURAL NETWORK OUTPUT VS. GROUND TRUTH WITH DIFFERENCE MAP: The deep learning framework is able to estimate the missing xy planes due to sparse z-sampling (gray) with data that looks similar to the ground truth for Ti-6Al-4V in (a) and Ti-7Al in (b) and (c). The misorientation angle map column shows the minimum possible misorientation between ground truth and estimated EBSD maps with 3 ° thresholded maximum to better show low magnitude errors. This map indicates that learning grain shapes for Ti-6Al-4V is more difficult than for Ti-7Al, likely due to smaller grain size and more grain boundary regions.	79

4.7	HISTOGRAM OF MISORIENTATION ANGLE FOR TI-6AL-4V, TI-7AL 1%, AND TI-7AL 3%: In (a), histograms of misorientation differences between predicted and ground truth are shown, where all values greater than 3° are clamped to 3° . For all materials, most network error in predicted misorientation is lower than 0.5° in magnitude. In (b), the same error histograms are displayed, but now misorientation values less than 3° are clamped to 3° . Because larger magnitude errors occur far less frequently than smaller errors, (c) contains a zoomed inset of misorientation angles greater than 3° to better show their distribution.	80
4.8	COMPARISON OF NETWORK OUTPUT WITH NEAREST NEIGHBOUR: Nearest neighbor interpolation method gives blocky grains and visually identical results to sparsely sectioned EBSD data.	83
4.9	NOISE IN ESTIMATED XY PLANE: The deep learning framework is able to estimate the missing xy planes in z dimension but there are some minor noises in some of the xy planes.	84
4.10	QUATERNION CNN AND REAL CNN: Both QCNN and RCNN have same number of layers except basic convolution operation layer. In QCNN, total number of trainable parameters are reduced significantly which give us room to add more complexity such self-attention layer to learn global features.	84
4.11	BISQUE UI INTERFACE A user interface of generating high-resolution 3D Microstructure from given sparsely sectioned 3D Microstructure	86
4.12	SPARSELY SECTIONED MICROSTRUCTURE: Visualization of a sparsely sectioned microstructure on BisQue	87
4.13	HIGH RESOLUTION MICROSTRUCTURE: Visualization of generated high resolution microstructure	87
4.14	DIFFERENCES BETWEEN REAL-VALUED CONVOLUTION AND QUATERNION CONVOLUTION: Real-valued convolution operations compute correlations, which are scalar quantities, between independent kernels and quaternion input feature maps. In contrast, quaternion convolution operations compute the Hamilton product, which is a vector quantity, between quaternion kernels and quaternion input feature maps. Furthermore, quaternion convolution requires only one-fourth of the kernel filters needed in real-valued convolution for the same number of output channels, resulting in a substantial reduction in total kernel filters.	92
4.15	POOLING AND CONCATENATION OF QUATERNION FEATURE MAPS: Following the first quaternion convolution layer with K_2 filters, as illustrated in Figure 4.14, the resulting feature maps are pooled into scalar and vector components (i^{th} , j^{th} , k^{th}) of Quaternion Feature Maps, with each component having a K_2 channel dimension. These pooled features are then concatenated along the channel dimension, resulting in a total of $4K_2$ channel dimensions.	93

4.16	EXPLANATION OF SECOND LAYER IN QUATERNION CONVOLUTION NEURAL NETWORK: In the second layer of the quaternion neural network, the Hamilton product is computed between the output of the pooling and concatenation operation, as shown in Figure 4.15, and K_3 quaternion kernel filters. This results in K_3 quaternion output feature maps, each with a dimension of $(4, H, W)$	94
5.1	GRAIN DISTRIBUTIONS ACROSS MATERIALS AND PROCESSES: 3D datasets collected from Ti-6Al-4V (top left), strontium titanate (top right), and additively manufactured (AM) Inconel 718 superalloy (bottom) samples with detailed examples of specific grains contained within. Data shown was gathered with 3D electron backscatter diffraction (EBSD). Average grain size is much larger relative to voxel size resolution in the strontium titanate dataset compared the Ti-6Al-4V dataset, resulting in fewer total grains, but better defined grain facets. The Inconel 718 sample and the grain contained within the melt pool on the top surface of the sample are elongated due to the fast cooling rates present in AM processes.	100
5.2	GENERATIVE ADVERSARIAL NETWORK (GAN) ARCHITECTURE OVERVIEW. A layout of the architecture of a traditional GAN, as described by [3]. . .	107
5.3	GENERATOR ARCHITECTURE OF M-GAN. A mapping network comprised of eight Fully Connected layers (FC) with Leaky ReLU activation function after each FC layer takes as input a 512 dimensional latent vector z . The output is then mapped to an intermediate latent space \mathcal{W} , converted into styles using a learned affine transformation (A), and passed through an AdaIN operation for each of the five blocks in the synthesis network. Block 1 passes constant input through AdaIN and ReLU activation functions, while Blocks 2-5 are deconvolution blocks progressively grown from $8 \times 8 \times 8 \rightarrow 64 \times 64 \times 64$	108
5.4	EXPERIMENTAL TRAINING SET: Ti-6Al-4V dataset, whose grains were used to train M-GAN on grain shape recognition. Sample is shown to scale in inverse pole figure (IPF) color. The IPF color indicates the orientation for each individual voxel, and these orientations are used in the segmentation process to identify individual grains, which are subsequently used for training.	111
5.5	SHAPENET OUTPUT. The M-GAN network generates detailed shape information for a diverse range of 3D objects. As the network learns from shape feature distributions, these objects are similar but not identical; such as the different styles of chairs shown in the bottom row. The variation within a class of generated objects that satisfies functional requirements is applied to the generation of grains across different material and processing classes as shown in Figure 5.1.	116

5.6	VISUAL COMPARISON OF GRAINS. Sampling of grains from real and generated sets are shown to verify visual similarity. Voxel size is $1.5 \mu\text{m} \times 1.5 \mu\text{m} \times 1.5 \mu\text{m}$. For generated grains, only largest connected component is shown. Stochastic nature of grains means no directly matching shapes are expected. Shape distributions are compared in Figure 5.7.	118
5.7	QUANTITATIVE EVALUATION OF M-GAN GRAIN GENERATION: Moment invariants ($\Omega_1, \Omega_2, \Omega_3$) are used to evaluate the similarity of real and generated shape distributions, as grains shapes are stochastic and correctness is a matter of shape probability rather than object recognition. Histograms are shown for both real (top) and generated (bottom) grain shapes. Vertical lines indicate the mean.	119
5.8	BISQUE MODULE FOR MGAN. This module generates a 3D single grain from a random noise vector.	121
5.9	Generated results for Rifle	126
5.10	SINGLE GRAINS. Nearest Neighbor Results	127
5.11	SHAPENET DATASET. Nearest Neighbor Results	128
5.12	SHAPENET DATASET. Nearest Neighbor Results	129
6.1	SYNTHETIC 3D MICROSTRUCTURE GENERATION PROCESS: In our framework, we want to generate 3D synthetic microstructure from given processing parameters which satisfies certain physical properties. In the first step, synthetic grains with crystal orientation information are generated using generative models from given parameters. In the second step, grains are encoded into latent space. In step 3, we learn optimal arrangement of grains using large language models in latent space. In step 4, latent spaces are decoded into grains and packed into a 3D volume to form 3D microstructure.	135

List of Tables

2.1	DESCRIPTION OF FILE STRUCTURE OF Ti-6Al-4V MICROSTRUCTURE DATASET	25
3.1	PSNR/SSIM COMPARISON FOR 4× SUPER-RESOLUTION SCALING: Physics-based loss consistently outperforms Bilinear, Bicubic, Nearest Neighbor, and pure L ₁ loss with no physics, regardless of architecture. Rows represent different loss functions and columns represent different network architectures. Higher number is desired for both PSNR/SSIM.	47
3.2	PSNR/SSIM COMPARISON FOR 4× ALGORITHMIC SCALING: PSNR/SSIM values for bilinear, bicubic, and nearest neighbour are consistently lower than network-based methods.	47
3.3	Percentage of Single Pixel Features: Rows represent different loss functions and columns represent different network architectures. Lower values indicate better performance, with ground truth containing approximately 0.2% single pixel features. Physics based loss reduces noise, leading to lower single-pixel feature counts.	47
3.4	Percent Feature Difference Over Input: Rows represent different loss functions and columns represent different network architectures. Lower values indicate better performance. Physics-based loss reduces noise and spurious features, which keeps the overall feature count closer to the amount expected based on input.	50
3.5	PSNR/SSIM COMPARISON FOR 4× SUPER-RESOLUTION SCALING ACROSS DIFFERENT MATERIALS: Ti-6Al-4V, Ti-7Al 1%, and Ti-7Al 3% for HAN Network: Physics-based loss consistently outperforms Bilinear, Bicubic, Nearest Neighbor, and pure L ₁ loss with no physics, and Columns represent different Titanium datasets. Higher number is desired for both PSNR/SSIM.	55

4.1	PSNR: COMPARISON OF PSNR OF MISORIENTATION ANGLE AND COMPLEXITY FOR DIFFERENT NETWORKS FOR SCALE FACTORS 2 AND 4: Columns represent number of trainable parameters and PSNR for different titanium datasets. A larger number is desired for both PSNR.	81
4.2	PSNR/SSIM of IPF Maps: COMPARISON OF PSNR/SSIM OF IPF MAPS AND COMPLEXITY FOR DIFFERENT NETWORKS FOR SCALE FACTORS 2 AND 4: Columns represent number of trainable parameters and PSNR/SSIM for different titanium datasets. A larger number is desired for both PSNR/SSIM.	85
4.3	PSNR/SSIM of IPF Maps: COMPARISON OF PSNR/SSIM OF IPF MAPS WITH AND WITHOUT QUATERNION TRANSFORMER FOR SCALE FACTOR 4: Columns represent PSNR/SSIM for different titanium datasets. A larger number is desired for both PSNR/SSIM.	85
5.1	Comparison of object-based performance for the ModelNet10 and ModelNet40 datasets using the feature extraction approach described by [4]. . .	117

Chapter 1

Material Microstructure in Process-Structure-Property

The quest for new materials is fundamental to scientific advancement and technological innovation. However, the traditional process of materials discovery and development is increasingly challenged by prohibitive time, labor, and capital demands. A particular bottleneck lies in the painstaking pace of experimentation, especially for crystalline materials. This thesis seeks to enhance this landscape through a transformative approach that leverages cutting-edge technologies and methodologies. Central to our transformative approach is the generating high-resolution microstructure data using physics based super-resolution methods to improve the microstructure data collection and generation of 3D synthetic grains with the objective of microstructures generation using deep-learning based generative models a strategy that holds immense promise for expediting the exploration of the intricate process-structure-property (PSP) relationships. By harnessing the power of high-fidelity 3D experimental microstructure datasets, we aim to forge an alternative pathway that not only streamlines but also enhances the process of discovering materials with specific and desired physical attributes.

1.1 3D Microstructure Data

Towards this, we suggest employing the creation of high-resolution microstructure datasets and 3D synthetic grain generation, fundamental components for producing 3D microstructures, as a key strategy to accelerate the study of process-structure-property (PSP) relationships. Central to our proposed effort is the integration of advanced technologies such as the TriBeam electron microscopy, an extensive repository of 3D material structure datasets with cutting-edge physics-based deep neural networks. The proposed methodology leverages state-of-the-art computational tools and deep learning algorithms to create highly accurate and detailed high-resolution microstructure datasets. These representations, borne from the integration of empirical data and advanced modeling techniques, offer a more efficient and cost-effective means of predicting material behavior and properties.

High resolution microstructure data is particularly attractive for tasks such as modeling mechanical or electrical properties, predicting variability in these properties, and characterizing rare and detrimental patterns of structure. However, existing statistically-based microstructure generators [5,6] face severe challenges related to generation speed and their ability to faithfully capture and statistically represent material structure. The recent development of rapid, high resolution 3D characterization techniques, combined with deep learning [7–11], provides a new pathway for synthetic microstructure generation through generative models, which have the potential to learn and produce the unique microstructures resulting from complex processing paths, at a reduced cost. The proposed effort leverages the unique UCSB TriBeam library of 3D material structures to explore this alternative deep learning design path with a cross-cutting team with expertise in computer vision, deep learning, software infrastructure, 3D multimodal characterization and materials design.

Remarkable strides have been made in deep generative models [12, 13], particularly in computer vision, where they have shown promise in understanding and interpreting complex visual data. Critical to this progress is the availability of extensive, high-quality datasets, for training these models. However, a notable challenge arises when it comes to the collection of 3D microstructural information - acquiring this type of data is not only more resource-intensive but also significantly more time-consuming compared to other forms of 3D data acquisition. An advantageous aspect of microstructure data lies in its intrinsic nature, where its formation and appearance are deeply rooted in fundamental physical relationships at multiple spatial scales. These relationships can be effectively harnessed and integrated into advanced generative models. Such integration not only bolsters the realism and accuracy of the generated outputs but also presents a strategic approach to addressing the challenges associated with data collection.

Our motivation for the methods presented in this dissertation comes from two material science problems: High-Resolution microstructure data collection using Tri-beam and 3D grain generation for creating synthetic 3D microstructure generation. Due to the physics and multimodalities of microstructure of dataset, these problems differ from the common computer vision problems that deal with natural images, and offer new opportunities for research at the intersection of material science and machine learning.

1.2 High-Resolution 3D Microstructure Data Generation

Gathering 3D material microstructural information is time-consuming, expensive, and energy-intensive. Acquisition of 3D data has been accelerated by developments in serial sectioning instrument capabilities; however, for crystallographic information, the electron

backscatter diffraction (EBSD) imaging modality remains rate limiting. We propose a physics-based efficient deep learning framework to reduce the time and cost of collecting 3D EBSD maps.

In the first segment of this work, we have successfully developed the first known super-resolution deep learning framework that operates on quaternion-based 3D crystal orientation data. This framework establishes a broadly-applicable pipeline for the super-resolution of microstructure data. It is particularly distinguished by its ability to generate network outputs that adhere to established scientific conventions in material research. Central to our framework is a physics-based rotational distance loss, leveraging crystal symmetry. We demonstrate that this physics-based loss outperforms traditional loss metrics such as L1 and conventional up-sampling algorithms, both qualitatively and quantitatively, across a variety of network architectures.

The second part of the work introduces a novel approach: the Quaternion Residual Block Self-Attention Network (QRBSA) paired with a physics-based loss function. This design is aimed at generating high-resolution 3D EBSD maps from sparsely sectioned EBSD maps. In QRBSA, quaternion-valued convolutions are adeptly employed to learn local relationships within the orientation space. Concurrently, self-attention mechanisms in the quaternion domain adeptly capture long-range correlations. We apply this framework to 3D data sourced from commercially significant titanium alloys. The results showcase, both qualitatively and quantitatively, our method's efficacy in predicting missing samples (EBSD information between sparsely sectioned mapping points) in comparison to high-resolution ground truth 3D EBSD maps.

Both components of this work will be useful in reducing the time and cost associated with microstructure data collection, particularly in settings utilizing Tri-beam setups. This advancement not only streamlines the process but also opens up new possibilities for efficient and accurate microstructural analysis.

1.3 Single Grain Generation for Synthetic 3D Microstructure Generation

Another approach of improving the cost and time is to generate synthetic 3D microstructure using generative models. The success of deep generative models depends on the availability of extensive, high-quality datasets for training these models. However, a notable challenge arises when it comes to the collection of 3D microstructural information - acquiring this type of data is not only more resource-intensive but also significantly more time-consuming compared to other forms of 3D data acquisition. An advantageous aspect of microstructure data lies in its intrinsic nature, where its formation and appearance are deeply rooted in fundamental physical relationships at multiple spatial scales. These relationships can be effectively harnessed and integrated into advanced generative models. Such integration not only bolsters the realism and accuracy of the generated outputs but also presents a strategic approach to addressing the challenges associated with data collection.

3D microstructural data consists of voxels, each encoding crystal orientation data, which are grouped into regions of similar orientations called grains. Beyond this, each pixel carries local crystal structure and chemical composition information, ultimately requiring multimodal information of disparate types for each pixel of a material. Generating individual grains with localized crystal orientations is less complex than synthesizing complete 3D microstructures through generative models.

This work explores the problem of generating 3D synthetic grains using the generative adversarial network (GAN). In this work, we propose a M-GAN network that is capable of recognizing and synthesizing individual grains whose morphology conforms to a given 3D polycrystalline material microstructure. This Generative Adversarial Network (GAN) architecture yields complex 3D objects from probabilistic latent space vectors with no

additional information from 2D images. We show that this method performs comparably or better than state-of-the-art on benchmark annotated 3D datasets, while also being able to distinguish and generate objects that are not easily annotated, such as grain morphologies. The value of our algorithm is demonstrated with analysis on experimental real-world data, namely generating 3D grain structures found in a commercially relevant wrought titanium alloy, which were validated through statistical shape comparison. This framework lays the foundation for the recognition and synthesis of polycrystalline material microstructures, which are used in additive manufacturing, aerospace, and structural design applications.

1.4 Summary of Contributions

This thesis makes significant contributions to the field of physics-based machine learning, with a specific focus on Electron Backscatter Diffraction (EBSD) super-resolution data. Our work on EBSD super-resolution includes developing an end-to-end deep learning training and inference framework specifically tailored for microstructure datasets. This framework, designed to address the 2D super-resolution problem, represents an innovative and effective approach to handling complex microstructural data. A key innovation in this research is the introduction of a physics-guided loss function for 2D super-resolution of microstructure datasets. This approach integrates domain-specific knowledge into the deep learning process, thereby enhancing the accuracy and relevance of the super-resolution outcomes.

Extending the above, we designed a physics-based neural network using a quaternion transformer network. This development is particularly noteworthy for its ability to generate 3D high-resolution microstructure datasets, demonstrating an advanced application of deep learning techniques in three-dimensional microstructural analysis.

Further, we present perhaps the first generative adversarial network (GAN) focused on learning grain morphology. This development is crucial for understanding and replicating the complex patterns found in granular materials. This innovative methodology opens new avenues in the generating 3D synthetic microstructures.

To support the research communities in computer vision/AI and material science, the method described above work has been developed into modules for BisQue [14](https://bisque2.ece.ucsb.edu/client_service/), a leading platform for bioimaging and material science. This integration aims to simplify access to complex materials data, removing obstacles related to data formats for computer vision/AI researchers. Simultaneously, it provides material scientists with advanced AI modules to enhance their experimental capabilities. This effort is intended to foster collaboration and innovation across disciplines by making cutting-edge tools more accessible to researchers.

1.5 Thesis Outline

In chapter 2, an overview of microstructure datasets is discussed. We introduce different microstructure modalities, the dataset we use in the dissertation and tools for visualization.

In chapter 3, we present a EBSD 2D super-resolution technique using a physics based deep learning framework. This chapter discusses quaternion-based orientation recognition loss functions that consider rotational effects and crystallographic symmetry, and an inference pipeline to convert network output into established visualization formats for EBSD maps. We apply this approach to EBSD data of a commercially relevant titanium alloy, and demonstrate both quantitatively and qualitatively that the generated results using physics-based loss are significantly better than those achieved using commonly used L_1 loss approaches or traditional upsampling algorithms such as bilinear, bicubic,

and nearest neighbor. The ability to generate physically accurate, high-resolution EBSD maps with super-resolution enables high throughput characterization and broadens the capture capabilities for three-dimensional experimental EBSD datasets.

In chapter 4, we provide extension of 2D EBSD super-resolution to 3D EBSD super-resolution using quaternion transformer network. We propose a physics-based efficient deep learning framework to reduce the time and cost of collecting 3D EBSD maps. Our framework uses a quaternion residual block self-attention network (QRBSA) to generate high-resolution 3D EBSD maps from sparsely sectioned EBSD maps. In QRBSA, quaternion-valued convolution effectively learns local relations in orientation space, while self-attention in the quaternion domain captures long-range correlations. We apply our framework to 3D data collected from commercially relevant titanium alloys, showing both qualitatively and quantitatively that our method can predict missing samples (EBSD information between sparsely sectioned mapping points) as compared to high-resolution ground truth 3D EBSD maps.

In chapter 5, we present a Generative Adversarial Network (GAN) capable of producing realistic microstructure morphology features and demonstrates its capabilities on a dataset of crystalline titanium grain shapes. Alongside this, we present an approach to train deep learning networks to understand material specific descriptor features, such as grain shapes, based on existing conceptual relationships with established learning spaces, such as functional object shapes. A style-based GAN with Wasserstein loss, called M-GAN, was first trained to recognize distributions of morphology features from function objects in the ShapeNet dataset, and was then applied to grain morphologies from a 3D crystallographic dataset of Ti-6Al-4V. Evaluation of feature recognition on objects showed comparable or better performance than state-of-the-art voxel-based network approaches. When applied to experimental data, M-GAN generated realistic grain morphologies comparable to those seen in Ti-6Al-4V. A quantitative comparison of mo-

ment invariant distributions showed that the generated grains were similar in shape and structure to the ground truth, but scale invariance learned from object recognition led to difficulty in distinguishing between the physical features of small grains and spatial resolution artifacts. The physical implications of M-GAN's learning capabilities are discussed, as well as the extensibility of this approach to other material characteristics related to grain morphology.

In the concluding chapter 6, we summarize the key contributions and discuss potential avenues for future work that can build upon the foundations established in this research. Generating individual grains with localized crystal orientations is a first step in synthesizing complete 3D microstructures through generative models. However, we require an advanced algorithm capable of constructing a continuous 3D synthetic microstructure from these grains, while adhering to the governing physical principles of grain boundary interactions, which combined with material processing steps, contribute to the grain morphology. In future work, we discuss how large language models can be used to construct 3D microstructure from single grains.

Chapter 2

Microstructure Datasets

In this chapter, we delve into the intricacies of microstructure datasets, exploring their various applications and the methodologies employed in their collection. We also address the challenges inherent in gathering such datasets, shedding light on the complexities and nuances involved in this process. This chapter is divided into the four subsections.

1. Introduction to Microstructure Datasets: This subsection contains a brief overview of microstructure datasets, elucidating their significance and widespread applications. A detailed examination of the techniques and strategies employed in the collection of microstructure datasets are discussed. We also discuss an in-depth analysis of the challenges faced during the collection of microstructure datasets.
2. Microstructure Dataset Representation: A detailed explanation of microstructure dataset representation in orientation space, and its properties such as symmetry space, fundamental zone and visualization of microstructure are discussed in this subsection.
3. Titanium Microstructure Dataset: We discuss the details of Titanium microstructure dataset such as Ti-6Al-4V and Ti-7Al dataset, which are strained plastically

to 1% and 3%. These datasets are commonly used in aerospace applications.

4. Accessing 3D Microstructure Dataset on BisQue: We have provided Ti-6Al-4V dataset that are openly accessible at BisQue [14, 15], a open source web based cloud infrastructure.
5. Conclusions of this chapter.

The contents of this chapter are from our published paper [15].

2.1 Microstructure Dataset

Most materials used for engineering applications are composed of an arrangement of elemental constituents into crystalline phases, which control the properties of that material. The arrangement of these crystals (also known as grains) is referred to as the microstructure, and is described using metrics that capture the size, arrangement, connectivity, and crystallographic orientation of the grains. Efficiently obtaining sufficient microstructure information to predict material properties is crucial to the development of new material composition and processing paths, especially for use in extreme environments. For both material and component design, 3D microstructure data plays a critical role in property prediction by informing the connectivity of grains and crystalline phases and characteristics of their interfaces for a broad range of applications, from biomedical to aerospace [16].

Collection Methods: Electron Back Scatter Diffraction (EBSD) [17] is a widely used scanning electron microscopy (SEM) technique that images a material surface and collects microstructure data that contains crystal orientation information. Microstructure data describes the arrangement of crystalline structures in a material, where “crystalline” refers to a collection of atoms exhibiting long-range order. For crystalline materials, the

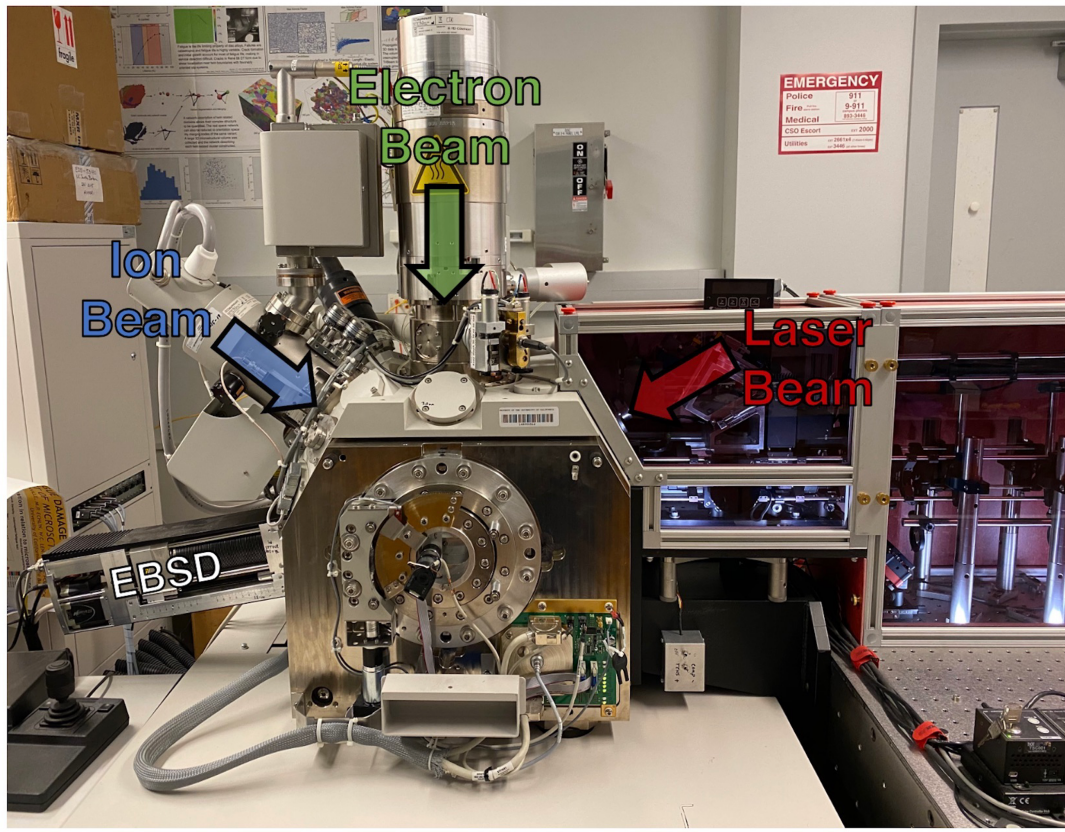


Figure 2.1: TRI-BEAM MICROSCOPE: 3D microstructure datasets are obtained during a serial sectioning experiment. The TriBeam is so named due to the presence of three beams: the electron, focused ion, and laser. The electron beam is used to gather diffraction patterns, whereas the laser beam is used to ablate the materials, layer-by-layer. The ion beam can be used clean the laser ablated surface in damage sensitive materials, if necessary.

arrangement of atoms significantly influences several material properties including yield strength, ductility, and fatigue resistance. Light optical microscopy and the resulting RGB images have inadequate resolution to image atoms or to evaluate atomic arrangement. Therefore, we rely on X-ray or electron diffraction for understanding the arrangement. During EBSD, a material is imaged one pixel at a time using a SEM, and electrons are diffracted from the atoms in the crystal according to Bragg's law. These diffracted electrons are collected on a detector producing a pattern called Kikuchi bands, which are indexed into crystal orientations to extract information about crystallographic ar-

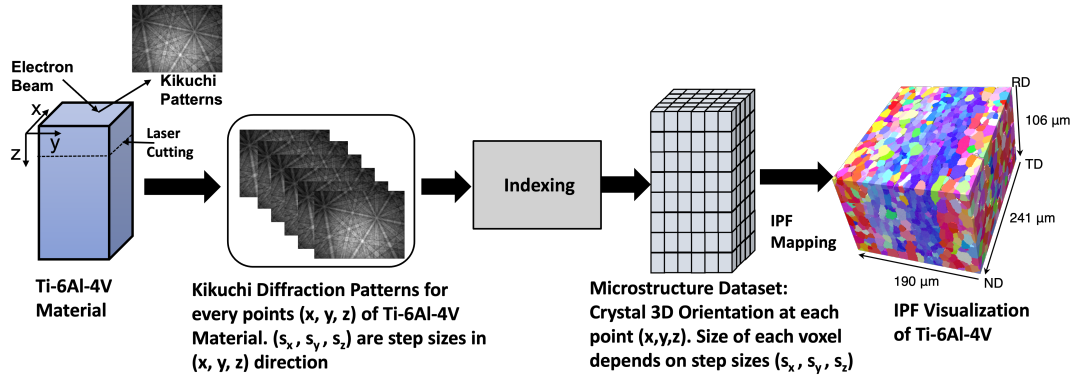


Figure 2.2: MICROSTRUCTURE DATA COLLECTION PROCESS: Material researchers collect Kikuchi diffraction patterns for each (x, y, z) coordinate of materials with the given step sizes (s_x, s_y, s_z) in (x, y, z) axis. The Kikuchi diffraction patterns are then indexed to determine the crystal orientation at each point. This information about crystal orientation is useful for predicting material properties. The inverse Pole Figure (IPF) technique is used to visualize the 3D crystal orientation.

rangement. There are several techniques for determining the crystal's orientation from a Kikuchi pattern, including mathematical approximations of band locations using Hough or Radon transforms [18,19], dictionary-based spherical cross-correlation approaches that leverage simulations of the electron-materials interactions [20,21], and network-based approaches [22]. The resultant data during EBSD is a diffraction pattern at each pixel in an image, which is indexed (mapped) into a crystal orientation represented as a vector (e.g. quaternion, Euler angle).

Challenges: One of the limitations of both light optical microscopy and EBSD is that they can only gather 2D images of materials that are fundamentally 3D. For both material and component design, 3D microstructure data plays a critical role in property prediction by informing the connectivity of grains and crystalline phases and characteristics of their interfaces for a broad range of applications, from biomedical to aerospace [16]. As a result, 3D microscopy techniques, like the TriBeam depicted in Figure 2.1, have been developed for this purpose. The TriBeam microscope is used to remove material in a layer-by-layer fashion from the sample using a femtosecond pulse

laser, capturing images at the surface of each slice using an electron beam and a suite of detectors [23]. These slices are then aligned, indexed, and assembled into a 3D volume following the pipeline process shown in Figure 2.2. The TriBeam method has made 3D microstructure collection more accessible [23,24], but even with enhancements in detector speeds and laser material removal rates, gathering this information remains expensive, energy-intensive, and time-consuming.

2.2 Microstructure Dataset Representation

The crystal orientations embedded in this microstructural information differ from the information in conventional light optical images, and are critical for developing and designing materials for a range of applications. The microstructure contains 3D crystal orientation information at each voxel, and 3D orientation can be represented in multiple ways, as discussed below.

2.2.1 Orientation Representation

The orientation of a 3D point can be represented in many ways, including Euler angles, rotation matrices, quaternions, axis-angle pairs, and Rodrigues vectors. Each orientation representation has distinct advantages and disadvantages in terms of ease of use for different calculations and data visualization, and there are packages available to readily convert between them [25,26]. Within these possible representations, quaternions are frequently used to avoid ambiguity in 3D rotations (gimbal lock) in tasks like software graphics, computer vision, and robotics. It has been demonstrated that quaternion representation is mathematically robust for crystal orientation representation and allows for easy application of symmetry operators between different crystal systems.

Rotation Matrices: Rotation matrices rotate a vector in 3 dimensions as a multi-

plication by a 3×3 matrix. With 3 degrees of freedom in a rotation, and 9 elements in the matrix, additional constraints need to be enforced on the matrix to make it a valid rotation. The rows must all be unit magnitude, mutually orthogonal, and the whole matrix must have a determinant of 1. Rotation matrices were not selected for this work as enforcement of these constraints would become difficult in a deep learning network. However, it should be noted that the composition of matrix rotations can be represented as a simple matrix multiplication.

Euler angles: Any arbitrary 3D rotation can be decomposed into three successive rotations around the coordinate axes. These angles can be defined as either extrinsic (rotations done using a fixed reference axis) or intrinsic (rotations done about an axis that is attached to the rotating body). Since there are three independent axes, denoted xyz , there are several possibilities for the decomposition across both extrinsic and intrinsic approaches, but here we will consider intrinsic only. By definition, intrinsic Euler angles must have matching first and third rotation axes, which, when applied across all possible xyz combinations, results in a total of six conventions for intrinsic angle representation. Here, we will be using Bunge convention, which has axis triple zxz with the corresponding rotation angles (ϕ_1, Φ, ϕ_2) (applied from left to right).

For Euler angles, there exists a fundamental ambiguity when expressing rotations that occur along coincident planes based on convention definition. For example, in Bunge convention, there is ambiguity associated with distinguishing between a $z00$ and a $00z$ rotation, regardless of crystal system. This ambiguity is commonly referred to as gimbal lock, and is seen in a wide variety of applications beyond crystal representation. The presence of gimbal lock makes Euler angles a less desirable choice as a network learning space, and is the reason we have chosen quaternion representation.

Quaternions: A quaternion q is a four component number of the form $q = q_0 + iq_1 + jq_2 + kq_3$, where imaginary units (i, j, k) satisfy the following relationship:

$$i^2 = j^2 = k^2 = ijk = -1 \quad (2.1)$$

Unit quaternions can always be written in the form

$$q = \cos \frac{\omega}{2} + \sin \frac{\omega}{2} (c_1 i + c_2 j + c_3 k) \quad (2.2)$$

where c_i are the directions cosines of the rotation axis unit vector \hat{n} . Unit quaternions are located on the sphere S_3 inside the 4D quaternion space. The 3D surface area of this sphere is equal to $2\pi^2$. The unit quaternion q and $-q$ represent the same rotation, so all rotations can be represented by selecting the entire northern ($q_0 \geq 0$) hemisphere, which has surface area of π^2

Quaternion representation is preferable due to its simplicity in computing orientation differences (misorientation) and the ease of enforcing constraints for a valid unit quaternion rotation. Furthermore, the only redundancy in quaternion representation is that $q = -q$, which is computationally trivial to address. The efficiency and lack of ambiguity in quaternion representations also make them well-suited to orientation expression in neural networks, both for loss functions [27] and network layer design [28]. We chose quaternions over other rotation representations for several reasons. The first is the ease in computing misorientation, which is the rotation operation which will transform one orientation into another. This misorientation can be computed using a quaternion multiplication. The second is the simplicity in enforcing the constraints to make a 4 dimensional vector a valid quaternion rotation. Here, it is only required that the vector be of unit magnitude.

Homochoric and Cubochoric Coordinates: Homochoric coordinates are a re-

duction of the quaternion representation space designed for a clearer visualization of material texture that avoids redundancies. In homochoric space, the northern hemisphere of quaternion space is projected onto a sphere in \mathbf{R}^3 , which removes quaternion redundancy. Cubochoric space further expands on this by transforming the homochoric sphere into an equal-volume cube, creating an orthogonal, equal-unit-area space for representation of material texture. Both of these approaches are discussed in more detail in [25, 29].

Axis-Angle Pairs: Axis angle pairs are a rotation description where the orientation of an object with respect to a reference frame is represented by a rotation of a set magnitude θ and a unit vector axis \mathbf{k} about which the rotation is made. For a given axis angle pair, (θ, \mathbf{k}) , the value a rotated vector \mathbf{v}' can be determined from the reference vector \mathbf{v} using Rodrigues rotation formula.

$$\mathbf{v}' = \mathbf{v} \cos \theta + (\mathbf{k} \times \mathbf{v}) \sin \theta + \mathbf{k} (\mathbf{k} \bullet \mathbf{v}) (1 - \cos \theta) \quad (2.3)$$

This approach can also be used with basis vectors to determine rotation matrices for various coordinate spaces.

Rodrigues Vectors: Rodrigues parameterization is a mathematically intuitive vector representation for orientations and rotational operations that take a form similar to axis-angle pairs. A given Rodrigues vector \mathbf{r} takes the form:

$$\mathbf{r} = \mathbf{k} \tan \left(\frac{\theta}{2} \right) \quad (2.4)$$

Where θ is the rotation angle, and \mathbf{k} is the axis of rotation, similar to axis angle pairs. Many different orientation relationships can be intuitively expressed in Rodrigues space, but because vector magnitude is scaled based on a nonlinear rotational relationship, many

traditional vector operations are not conserved in their classical forms, so care must be taken to perform vector addition and multiplication correctly. Additional derivations and relationships of Rodrigues space are discussed in [30].

Grains: Grains are the region in microstructure that have similar crystal orientation. Polycrystalline materials like Ti-6Al-4V consist of a large number of subdomains called grains, which are visible as regions of relatively uniform crystallographic orientation (uniform color) in Figure 2.3.

2.2.2 Symmetry Space

Optically-gathered images commonly used in computer vision tasks encode color and intensity information as scalars at each pixel. On the other hand, crystallographic data encodes orientations as vector-based rotations relative to a chosen reference frame. The range of possible unique crystal rotations is constrained by the symmetry of the crystal, such as body centered cubic (BCC), face centered cubic (FCC), or hexagonal close packed (HCP). Symmetry has a disruptive impact on learning with standard loss functions (L-norms) because symmetry planes create duplicates and discontinuities in rotation distance measurement. Multiple-defined rotations are avoided through the fundamental zone convention, which is defined as the polyhedron in Rodrigues space that encompasses all angles whose distance is closer to the 0 rotation than to any symmetric equivalent of the 0 rotation [30]. The Ti-6Al-4V, Ti-7Al 1%, and Ti-7Al 3% microstructure datasets released here is an HCP material (space group 194, point group $6/mmm$), so it has a total of 24 symmetry operators, of which only 12 do not involve a change of handedness.

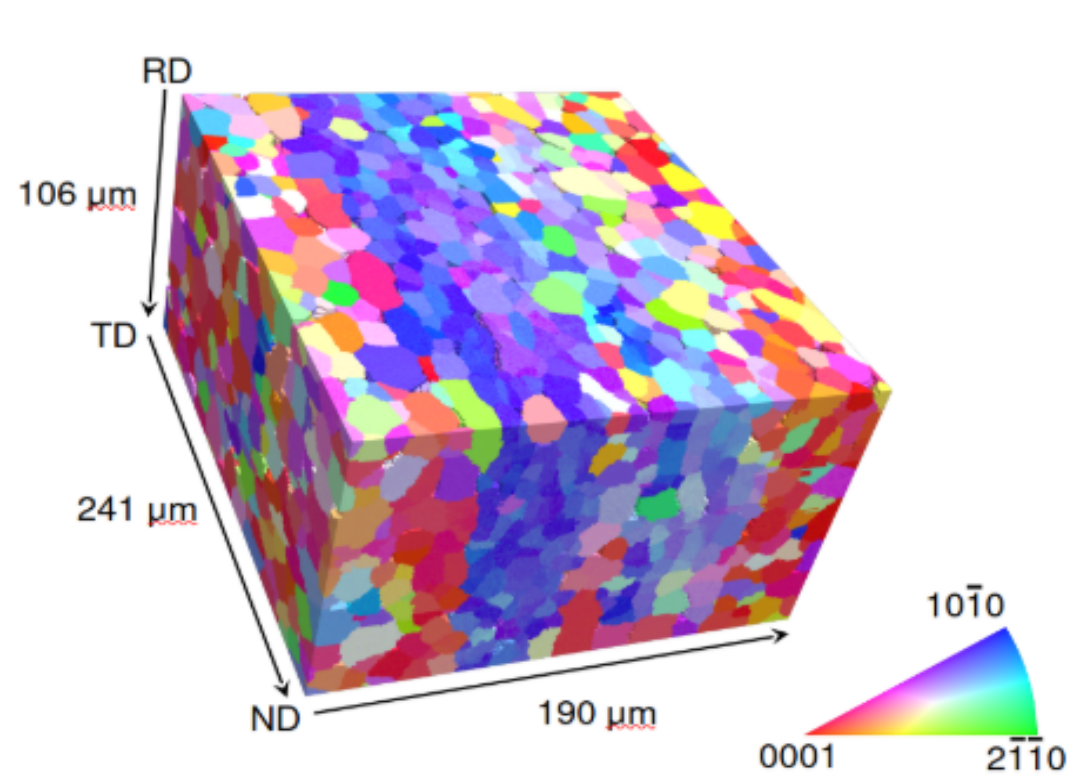


Figure 2.3: INVERSE POLE FIGURE (IPF) MAP OF TI-6AL-4V-EQUIAXED MICROSTRUCTURE DATASET: The color represents the orientation of crystal. Each grain of the same color has a nearly identical crystallographic orientation. Rolling direction (RD), transverse direction (TD) and normal direction (ND) reference arrows are also shown. IPF coloring is referenced to the TD.

2.2.3 Fundamental Zone Reduction

Given the symmetric nature of crystals, there are often many rotations which can represent a single crystal structure. The group of rotations depend upon the type of crystal structure of the material. Given the many equivalent rotations, fundamental zone reduction chooses the rotation which has the smallest rotation angle.

This idea of a minimum possible rotation can be formalized mathematically. In Rodrigues space, it is intuitive to describe a minimum unique region that fully encloses all possible orientations for vectors following the constraints of a given symmetry group. This space is known as the fundamental zone, and it can intuitively be thought of as

the minimum range of rotation space needed to describe all possible orientations of a crystal when accounting for symmetry. The formal definition of the fundamental zone, as described by [30] is:

$$\bigcap_{i=2}^N \left\{ \mathbf{r}; \tan \left(\frac{\theta_i}{4} \right) \pm \mathbf{r} \bullet \mathbf{l}_i \geq 0 \right\} \quad (2.5)$$

where \mathbf{l}_i and θ_i are the unit vector and rotation angle for the i th element of the symmetry group. \mathbf{r} represents a rotation by an angle ω about an axis given by vector n . N is order of the group, and the identity is assumed to correspond to $i = 1$. Fortunately, the relationship between quaternion and Rodrigues space allows for this calculation to be done rather straightforwardly directly on quaternions. In quaternion space, the formal definition described by [30] can be found by expanding all quaternions based on the rotations in the symmetry group and then reducing to only the quaternions with the largest positive scalar value within the expanded group.

2.2.4 Visualization of Crystal Orientations

When orientation data is produced, real or synthetic, a means of visualization is required, which is challenging, since many orientation representations (quaternion, rotation matrix, Rodrigues vector) do not directly translate into common image formats. The most direct option is to convert orientation output into a 3-channel form and map it directly to a color scheme (RGB, HSV, YUV), but the presence of symmetry means small orientation changes often result in large changes in color scale in this type of mapping, which can create the appearance of noise even when data is correct. For this reason, the Inverse Pole Figure (IPF) color scheme was designed, which stereographically projects the fundamental zone into 2D and maps a uniform RGB gradient onto it. An IPF legend for HCP is shown in the bottom right of Figure 2.3, and details of projection and color pattern choice are discussed in [17, 20, 31]. Note that even though this output is intu-

itive to visualize, it is a projection that is fundamentally ambiguous and not information preserving, so no learning or inference should be done on IPF images, as they cannot be converted back into orientations.

2.3 Titanium Microstructure

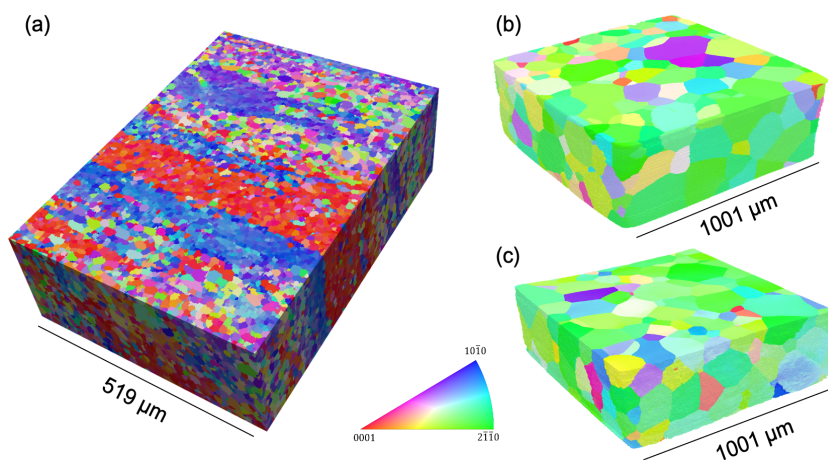


Figure 2.4: RENDERING OF 3D EBSD DATASET INVESTIGATED: shown in IPF coloring of the titanium alloys, (a) Ti-6Al-4V and (b) Ti-7Al mechanically loaded in tension to 1% strain and (c) 3%, used for network training, testing and validation. Dataset details are available elsewhere [1, 2].

The microstructure datasets of titanium alloys such as Ti-6Al-4V and Ti-7Al are illustrated in Figure 2.4. These alloys are commonly used in aerospace applications.

The ground truth data is experimental 3D microstructure (EBSD) data gathered from the titanium alloys, Ti-6Al-4V and Ti-7Al (one Ti-7Al sample deformed in tension to 1% and one to 3%), using a commercially-available rapid-serial-sectioning electron microscope known as the Tribeam [32, 33]. The Ti-6Al-4V dataset, shown in Figure 2.4(a), is of pixel size $346 \times 142 \times 471 \times 4$ ($z \times y \times x \times ch$), where the last dimension is the quaternion component. Analogously, the Ti-7Al shown in 2.4(b) and 2.4(c) are of size $232 \times 674 \times 770 \times 4$ ($z \times y \times x \times ch$) and $224 \times 770 \times 770 \times 4$ ($z \times y \times x \times ch$) pixels

respectively, with all edges cropped to produce a perfect parallelepiped volume. Each voxel in the Ti-6Al-4V set has resolution of $1.5 \times 1.5 \times 1.5 \mu\text{m}$, and in both Ti-7Al sets, each voxel has a resolution of $1.3 \times 1.3 \times 1.3 \mu\text{m}$.

These titanium alloys are composed primarily of the hexagonal close packed grains. The EBSD ground truth data was indexed using a combination of EMsoft dictionary indexing (Ti-7Al 1% and 3%) [20] or spherical indexing (Ti-6Al-4V) [21] for improved indexing accuracy beyond conventional Hough transform indexing. Experimental data was cleaned using a minimum size filter of 27 voxels in volume and a minimum feature neighbor filter of 2 neighbors per grain (applied using DREAM.3D [6]) to eliminate grains that were insufficiently or inaccurately resolved by the 3D characterization technique.

In total, the Ti-6Al-4V dataset contains about 57,000 grains, visible in the IPF maps as regions of different color. The Ti-7Al material has larger grain size, with 500-1000 grains in each dataset.

2.4 Accessing Microstructure Dataset on BisQue

In this section, the details of Ti-6Al-4V equiaxed microstructure dataset, available on BisQue [14, 15], are discussed. The physical dimension of the Ti-6Al-4V microstructure sample is $190 \times 241 \times 106$ micrometers. In voxel-based representation, this 3D volume expands to $318 \times 377 \times 121$ voxels, where each voxel contains information about the crystal orientation. All the relevant information for designing machine learning models relevant to this dataset is stored in `DataContainers/ImageDataContainer/CellData`, as shown in the Figure 2.5. Each grain in the dataset is composed of a collection of neighboring voxels that have similar crystal orientations. In the dataset itself, each grain is labeled with a unique positive integer value, termed a feature ID, that is randomly assigned, starting at 1 (the integer value 0 is typically reserved for any void regions cap-

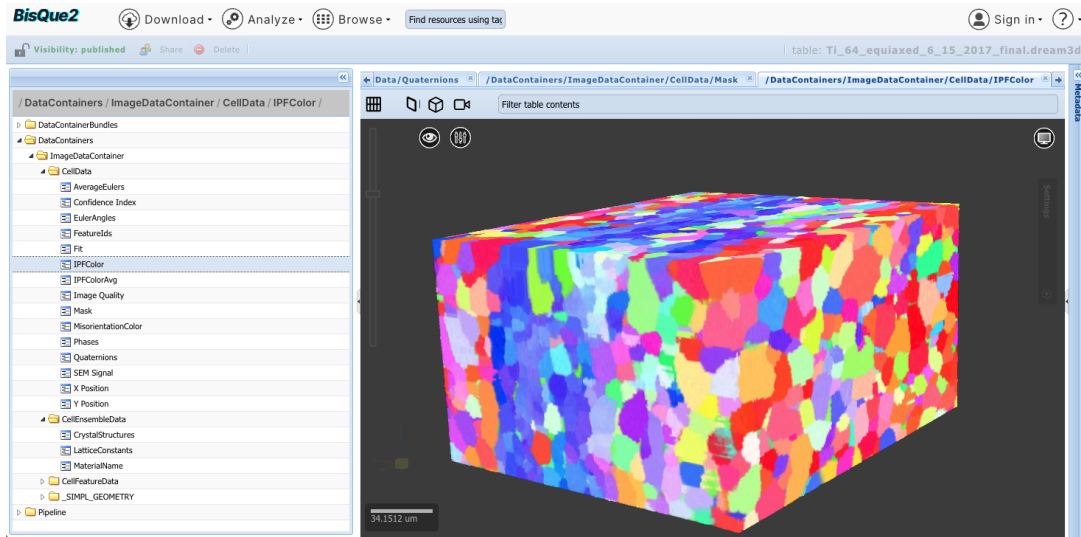


Figure 2.5: TI-6AL-4V MICROSTRUCTURE DATASET ON BISQUE: Ti-6Al-4V 3D microstructure datasets can be accessed on our open source web-based architecture "BisQue". The microstructure datasets are saved as hdf/dream3d format and can be visualized using "IPFColor" field. The crystal orientations at each voxel is saved in "EulerAngles" and "Quaternions" fields. "FeatureIds" field shows the location of grains, and each grain has a distinct feature id.

ured in the dataset). There are a total of 8893 single grains in the $318 \times 377 \times 121$ volume in voxel space, and of those, 6645 grains are completely captured such that they do not come into contact with the edge of the collection volume. Separating fully captured grains from edge grains is not usually necessary for voxel-based vision tasks, but is important for morphological studies where flat surfaces created by sample edges can lead to bias. The exact number of datapoints in the dataset will vary depending on the features of interest being studied. Any voxel level information will be stored in an array of $318 \times 377 \times 121 \times [\text{array depth}]$. Quaternion information, for example, is collected at the voxel level, so the array of quaternions in this dataset is of size $318 \times 377 \times 121 \times 4$. However, there is also information that can be collected at the grain level (also referred to as feature level), such as the average orientation in each grain, or the volume of each grain. Values like these would be stored as a list, with the value for a given grain stored at the list index that matches its integer feature ID.

The EBSD dataset presented in this work was processed using Dream3D [6], so the structure of the set follows the HDF5 architecture and nomenclature conventions used for Dream3D files. Files with the .dream3d extension are directly compatible with applications and packages used for HDF files (e.g. the HDF View software package or h5py in Python), so these can readily be used to extract individual variables from the file for independent manipulation. The file structure can also be visualized directly within the web-based BisQue infrastructure for convenience. A visualization of the file directory from BisQue can be seen in Figure 2.6, and a breakdown of variable definitions is available in Table 2.1.

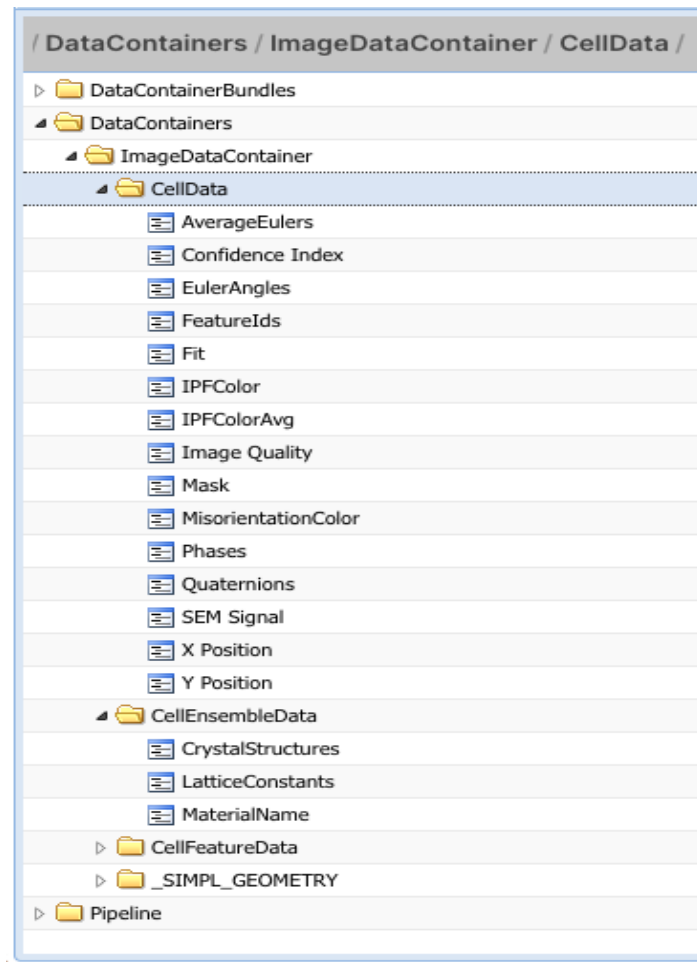


Figure 2.6: FILE STRUCTURE OF TI-6AL-4V MICROSTRUCTURE DATASET ON BISQUE .

CellData	
AverageEulers	The average Euler angle for each voxel's corresponding grain (See AvgEulers).
Confidence Index	The confidence value for the Hough Transform indexing used to determine orientation.
Euler Angles	Crystal orientation at each voxel using Bunge Euler angle representation.
FeatureIDs	Integer identifier values for each grain. Void is feature 0. Others assigned randomly.
Fit	Comparative quality metric between the indexed orientation and Hough transform (in degrees).
IPFColor	RGB color mapping of crystal orientation using IPF projection
IPFColorAvg	IPF projection mapping where each grain is colored by average orientation.
Image Quality	Image quality metric. Larger values indicate better quality.
Mask	Mask indicating regions of void. Void = 0, Solid = 1 (note: dataset contains no void).
Misorientation Color	RGB color representation for relative misorientation, developed by [34].
Phases	Voxel level numerical phase map. 1 = HCP α phase. 2 = BCC β phase.
Quaternions	Crystal orientation at each voxel using quaternion representation.
SEM Signal	Dataframe for storing equivalent SEM signal. Dataset is purely EBSD, so this is empty.
X Position	The x-position of each voxel in the volume
Y Position	The y-position of each voxel in the volume.
CellEnsembleData	
CrystalStructures	Dream3D internal value. Numerical labels for void, α , and β crystal structures.
LatticeConstants	Lattice constants for associated labels in CrystalStructures
MaterialName	String name labels for phases in CrystalStructures and LatticeConstants.
CellFeatureData	
Active	Internal label indicating which features are captured within the current dataset.
AvgEulers	Average Euler angle orientation of each grain feature.
AvgQuats	Average quaternion orientation of each grain feature.
Centroids	Centroid of each grain in physical space (see SIMPL_Geometry).
EquivalentDiameters	Diameter in μm of each grain if it were approximated as a sphere of equivalent volume.
NumCells	Number of voxels in each grain feature.
NumNeighbors	Number of adjacent feature neighbors for each grain.
Volumes	The volume of each grain in μm^3
._SIMPL_Geometry	
Dimensions	Size of the dataset in voxels.
Origin	Location of the physical space origin in in μm .
Spacing	Physical x, y, and z size of each voxel in μm

Table 2.1: DESCRIPTION OF FILE STRUCTURE OF Ti-6Al-4V MICROSTRUCTURE DATASET

2.5 Dream 3D Pipeline

The following pipeline has been used to construct these datasets. The details about the pipeline steps are available at <https://bit.ly/Ti6Al4Vmicrostructure>

Read H5EBSD File → Threshold objects → Convert Orientation Representation → Align Sections → Segment Features → Find Feature Size → Minimum Size → Find Feature Neighbors → Minimum Number of Neighbors → Find Feature Average Orientations → Fill bad data → Generate IPF Colors → Generate Misorientation Colors → Create Element Array from Feature Array → Generate IPF Colors → Find Feature Centroids → Crop Geometry (Image) → Write DREAM.3D Data File

2.6 Conclusion

In this chapter, we provide an overview of microstructure datasets, examining their acquisition and diverse applications. Additionally, we discuss the inherent challenges associated with collecting such datasets. We have made the Ti-6Al-4V equiaxed microstructure dataset available on the BisQue platform, offering a valuable resource to researchers and practitioners in the field.

Chapter 3

Adaptable Physics-based Super-resolution for Electron Backscatter Diffraction Maps

In this chapter, we delve into enhancing the resolution of 2D microstructure datasets through the application of physics-informed machine learning. Building upon the previous chapter, where we explored the challenges of acquiring high-resolution experimental 3D microstructures—namely the significant investment of time and financial resources—this section shifts focus to a pivotal question: Is it possible to employ a physics-guided machine learning approach to refine the resolution of microstructure datasets?

In computer vision, single image super-resolution (SISR) has been extensively explored using convolutional neural networks (CNNs) on optical images, but images outside this domain, such as those from scientific experiments, are not well investigated. Experimental data is often gathered using non-optical methods, which alters the metrics for image quality. One such example is electron backscatter diffraction (EBSD), a materials characterization technique that maps crystal arrangement in solid materials, which pro-

vides insight into processing, structure, and property relationships. We present a broadly adaptable approach for applying state-of-art SISR networks to generate super-resolved EBSD orientation maps. This approach includes quaternion-based orientation recognition loss functions that consider rotational effects and crystallographic symmetry, and an inference pipeline to convert network output into established visualization formats for EBSD maps. The ability to generate physically accurate, high-resolution EBSD maps with super-resolution enables high throughput characterization and broadens the capture capabilities for three-dimensional experimental EBSD datasets.

To address this, we have developed a novel framework incorporating a physics-based loss function. This approach has been rigorously tested across various network architectures to facilitate the transformation of low-resolution datasets into high-resolution Electron Backscatter Diffraction (EBSD) datasets. The following segments of this chapter are dedicated to presenting the methodology, experimental setup, and the results of these endeavors, providing insights into the efficacy and potential of physics-informed machine learning in enhancing microstructural data resolution. The contents of this chapter are discussed in our published paper [27].

3.1 Introduction

The term image super-resolution is used to describe methods designed to infer high-resolution (HR) image output from low-resolution (LR) input. Since their development, super-resolution methods have been used in applications such as surveillance and security, biometric information identification, remote sensing, astronomy, and medical imaging [35]. Generally, these algorithms can be categorized into three groups based on the information available during training: a) supervised, which have paired LR-HR images during training, b) semi-supervised, where no LR-HR image pairing is available, and

c) unsupervised, where no ground truth HR is available. Recently, because of their superior performance to traditional methods, various supervised deep CNN architectures using recursive residual blocks [36,37], residual connections, and attention-based modules [38–41] have seen significant use in super-resolution applications.

The approaches for different super-resolution methods vary, but they all share the common goal of producing high-resolution image output that is a clear representation of the low-resolution input in the context of both image content and visual fidelity. Generally, evaluation metrics are centered around the idea that the output image is the product of intensity-based visible-light photography, where the goal is to represent what is seen with the human eye. But for scientific image applications, this idea is often incorrect, since many experimental methods construct images or image maps using electromagnetic information gathered from outside the visible light range (e.g. X-rays, infrared information), or even not from light at all (e.g. electrons, neutrons), which means the ideas of sharpness and visual clarity have very different meanings. One such example, electron backscatter diffraction (EBSD), used for characterization of crystalline materials, relies on electron diffraction to build maps of material crystallographic information.

3.1.1 EBSD Imaging

EBSD is a scanning electron microscopy technique that maps crystal lattice orientation by analyzing Kikuchi diffraction patterns that are formed when a focused electron beam is scattered by the atomic crystal structure of a material according to Bragg’s law. A grid of Kikuchi patterns is collected by scanning the electron beam across the sample surface. These patterns are then indexed to form a grid of orientations, which are commonly represented as images in RGB color space using inverse pole figure (IPF) projections. EBSD datasets are used to determine microstructural properties of materials

such as texture, orientation gradients, phase distributions, and point-to-point orientation correlations, all of which are critical for understanding material performance [42]. Furthermore, the fact that EBSD datasets are maps rather than just images changes both the notion of image quality and the motivation for super-resolution methods.

EBSD has two types of resolution: the accuracy with which the EBSD pattern collected at each pixel can be indexed into a crystallographic orientation, and the spacing between pixels in a given mapping. During EBSD mapping, the electron beam must dwell at each point long enough to form a high quality Kikuchi pattern, which is then indexed into a given crystallographic orientation [43–45]. The indexing problem has many salient issues associated with it, among them the differentiation of matrix and precipitate phases, the identification of local strain effects, and the decoupling of overlapping diffraction patterns at grain boundaries. While accurate indexing is critical to EBSD, it is an independent challenge unrelated to super-resolution, as each indexing problem is treated as having no correlation with its neighbors. The lack of assumed spatial correlation separates indexing from other pixel-based problems and makes it ill-suited for SISR. Therefore, the issues associated with indexing accuracy described above are not addressed here. Instead, we consider the improvement of spatial resolution, which, in experiment, equates to collecting a higher density of data points during mapping. This requirement can lead to long mapping times or force the choice to use a coarser resolution mapping grid for expediency. The necessity to reduce EBSD collection time becomes even more critical when performing many scans during serial sectioning 3D EBSD measurements [33], where material is removed layer-by-layer with 2D maps collected at each slice, and then stacked into a 3D dataset. In almost all serial sectioning experiments, the minimum slicing thickness/resolution is much lower than the achievable in-plane imaging resolution, creating anisotropic voxels. Furthermore, poor electrically conductive materials become electrostatically charged by the beam or degrade from beam exposure

(e.g. bio materials, polymers), requiring extremely short exposure times and resulting in Kikuchi patterns with weak contrast.

As the demand for greater volume and more detailed resolution material information grows, so too does the demand and expense of EBSD mapping. Spatial resolution in EBSD is of particular interest in the characterization of deformed materials and additive manufacturing [46], where subgrain misorientation gradients are used to quantify local plastic deformation effects and geometrically necessary dislocation densities [47–49]. In efforts to improve EBSD resolution and quality, simulations and experimental studies [50–54] have shown that lowering the electron beam accelerating voltage can significantly improve the spatial resolution of EBSD maps, but map quality and achievable resolution vary with differing materials and imaging conditions. To improve indexing accuracy, multiple algorithmic approaches have been developed for better Kikuchi pattern mapping [20,21], which improves both the precision and accuracy of the orientations shown at each pixel. Machine learning approaches have also been used to accelerate several tasks in the EBSD map construction process, including Kikuchi pattern indexing [22], classification [55], and crystal identification [56]. Recently, a residual based neural network with traditional L_1 loss (ResNet) was used to produce super-resolved EBSD maps from inverse pole figure (IPF) color and Euler angles as an image input [57]. The desire to accelerate and improve the EBSD mapping process has motivated a wide array of machine learning approaches, but many challenges still exist. One of the most prominent of these is that orientation space is discontinuous and repeating, and the fundamental shape of orientation space changes with the symmetries of the crystal being observed. This makes brute-force network learning with traditional methods highly dependent on the available data for training, and, depending on the orientation, small variations in accuracy can produce dramatically incorrect results.

Given these challenges, we present an adaptable framework for neural-network-based

super-resolution of EBSD maps, where all network learning is built around the physics of crystal orientation symmetry. We define a physics-based loss that accounts for crystallographic symmetries, which is used alongside either a traditional L_1 loss metric or a loss based directly on rotational arc lengths, which correspond to conventional misorientation measurements in crystallography. All super-resolution is done on crystal orientation data expressed as quaternions, meaning each pixel in a given map contains four channels. Using quaternion space allows for complete representation of orientation space, enabling a training approach that is translatable across all 230 crystallographic space groups. As a proof of concept, four state-of-the-art residual and channel-attention networks are used to generate high-resolution EBSD maps from low-resolution input using this physics-based approach. We demonstrate that regardless of network choice, physics-based approaches outperform traditional approaches both qualitatively and quantitatively. This approach has direct application to experimental EBSD measurements of electron beam-sensitive or poor-conducting materials where charge buildup and extended beam exposure are limiting factors, and for 3D EBSD data collection where high out-of-plane imaging resolution is costly. We expect EBSD-SR to accelerate EBSD mapping for defect detection and fast screening of microstructure configurations that limit material properties.

3.2 Method

In this section, we begin by exploring the representation of datasets in orientation space, along with the preprocessing applied to both ground truth and low-resolution Electron Backscatter Diffraction (EBSD) maps. Subsequently, we delve into the specifics of the network architecture, including details on loss functions, the inference pipeline, implementation details of the network and hyper-parameters, and the methodology for evaluating outputs.

3.2.1 Orientation Representation

The focus of this investigation centers on orientation vector maps that describe crystalline domains which are fundamentally anisotropic and periodic in nature. Orientations for each pixel within the network learning environment are expressed in terms of quaternions of the form $q = q_0 + \hat{i}q_1 + \hat{j}q_2 + \hat{k}q_3$. For orientation representation, quaternions are beneficial due to their lack of ambiguity with respect to orientation representation and crystal symmetry. To avoid redundancy in quaternion space (between q and $-q$), all orientations are expressed with their scalar q_0 as positive. For visualization according to established conventions, quaternions are reduced to the Rodrigues space fundamental zone based on space group symmetry, converted into Euler angles, and projected using inverse pole figure (IPF) projection using the open-source Dream3D software [6].

3.2.2 Data Preprocessing

High-Resolution (HR) Ground Truth: The ground truth data is experimental 3D EBSD data gathered from two titanium alloys, Ti-6Al-4V and Ti-7Al, using a commercially-available rapid-serial-sectioning electron microscope known as the Tribeam [32,33]. The Ti-6Al-4V dataset, shown in Figure 3.5(a), is of pixel size $346 \times 142 \times 471 \times 4$, where the last dimension is the quaternion component. Analogously, the Ti-7Al shown in Figure 3.5(b) and 3.5(c) are of size $770 \times 674 \times 132 \times 4$ and $770 \times 770 \times 224 \times 4$ pixels respectively, with all edges cropped to produce a perfect parallelepiped volume. Each voxel in the Ti-6Al-4V set has resolution of $1.5 \times 1.5 \times 1.5 \mu\text{m}$, and in both Ti-7Al sets, each voxel has a resolution of $1.3 \times 1.3 \times 1.3 \mu\text{m}$.

These titanium alloys are composed primarily of the hexagonal close packed grains. In total, the Ti-6Al-4V dataset contains about 57,000 grains, visible in the IPF maps as regions of different color. The Ti-7Al material has larger grain size, with 500-1000

grains in each dataset. The datasets were proportionally divided into training, validation, and test subsets in ratios of 65%, 15%, and 20% respectively. The sample volume was sectioned such that each subset contains an equivalent fraction of orthogonal images from each face of the sample to avoid any bias due to material anisotropy. During training, each volume was broken into two-dimensional patches of size 64×64 pixels.

Low-Resolution (LR) Input: In the context of EBSD maps, we do not employ blurring filters as is commonly practiced with images. This is due to the nature of collecting diffraction patterns at each point using an electron beam, which does not interact with adjacent pixels. Consequently, instead of applying a blurring filter before downsampling, we simply remove rows and columns with a downscale factor to reflect how EBSD resolution would be reduced in actual experiments. This is done to imitate the beam raster steps that would occur in an EBSD experiment with lower resolution, where a lower resolution would not influence the electron beam-material interaction volume at each pixel, but rather lead to greater raster distance between consecutive pixels of the same size.

To find the maximum downsampling factor, we have computed the maximum frequency component of our datasets. To perfectly reconstruct the signal, the Nyquist sampling rate should be at least two times the highest frequency component of datasets. However, the highest frequency component for the given datasets is larger than the Nyquist sampling rate, therefore, we can not perfectly reconstruct the original signal after the downsampling operation.

3.2.3 Neural Network Architecture

Unlike traditional SR architectures that focus on 1 or 3-channel image data, the deep-learning framework developed in this study, shown in Figure 3.6, directly operates

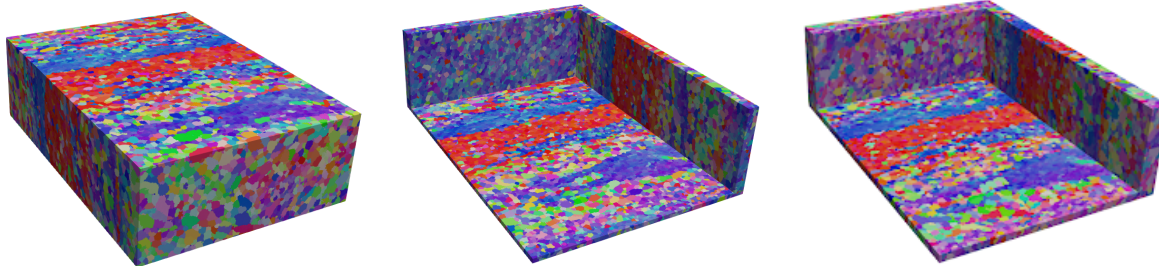


Figure 3.1: Training Set

Figure 3.2: Validation Set

Figure 3.3: Test Set

Figure 3.4: PRE-PROCESSING OF TI-6AL-4V DATASETS. Figure 3.1 has size of $300 \times 124 \times 408 \times 4$. Figure 3.2 and Figure 3.3 are divided into 3 mutually orthogonal parallelepiped blocks, which are then sliced into images.

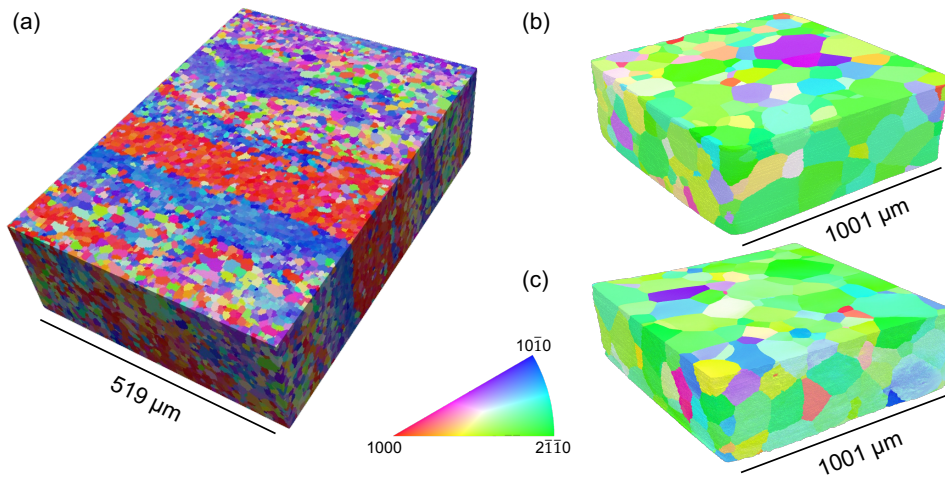


Figure 3.5: RENDERING OF 3D EBSD DATASET INVESTIGATED: shown in IPF coloring of the titanium alloys, (a) Ti-6Al-4V and (b) Ti-7Al mechanically loaded in tension to 1% strain and (c) 3%, used for network training, testing and validation. Dataset details are available elsewhere [1, 2].

on vector-based orientation data, which is expressed as 4-channel quaternions. Quaternions are mathematically robust for crystal orientation representation and allow easy application of symmetry operators between different crystal systems.

The process for this framework follows similar methodologies to experimental EBSD mapping, where *a priori* crystal symmetry information is used to inform the training and inference processes. For the physics-informed framework, a loss function satisfying

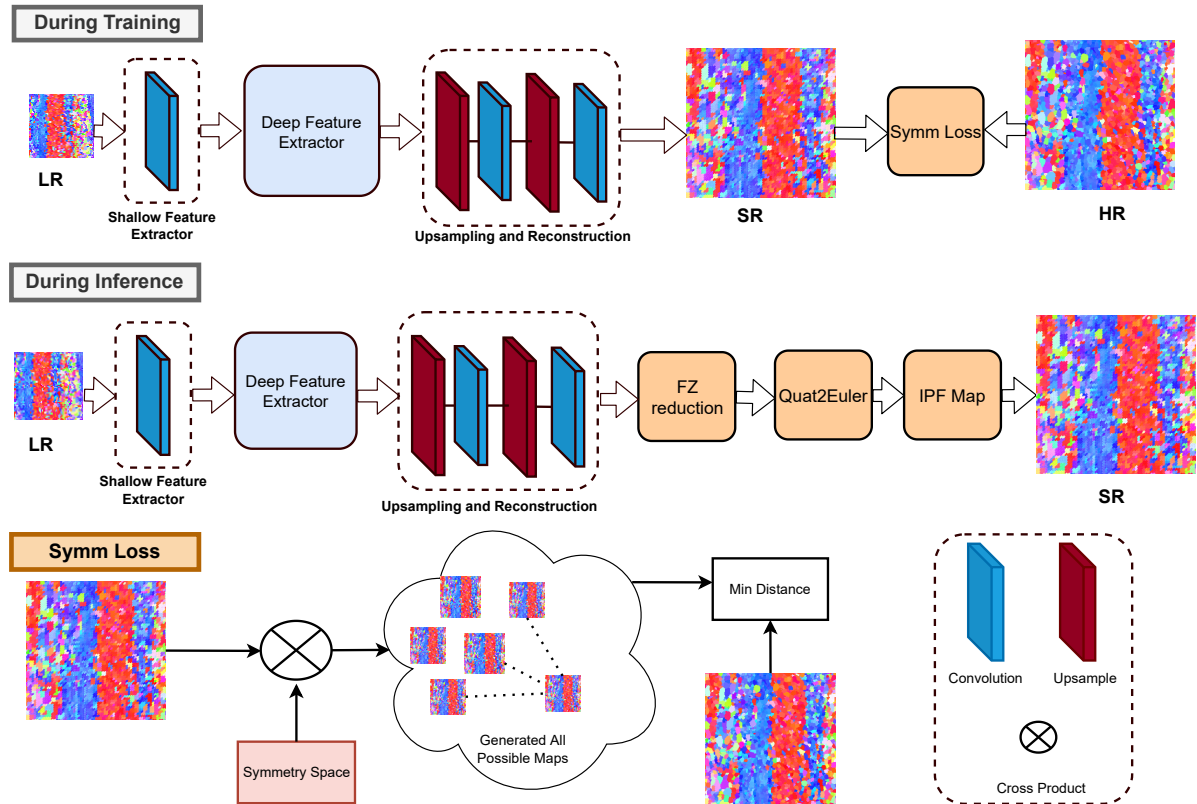


Figure 3.6: NETWORK ARCHITECTURE: **Training Pipeline:** A low-resolution EBSD map in quaternion orientation space is given to an image super-resolution network architecture that generates a high-resolution EBSD map in quaternion orientation space. A crystallographic symmetry physics-based loss with L_1 or approximate rotational distance is used during training. **Inference Pipeline:** The image super-resolution network generates a high-resolution EBSD map in quaternion orientation space, which is reduced to a fundamental zone space, and converted to Euler orientation space to visualize in IPF color map. **Symmetry Loss:** Takes all possible hexagonal symmetries for the titanium alloy, and computes the minimum distance between all possible generated output and ground truths. The distance can be L_1 or approximate rotational distance.

space group symmetry requirements is defined, and networks are trained with that loss function on datasets of materials from the corresponding space group. Once trained, the network can infer on EBSD maps for materials that fall under the same space group symmetry. The full framework consists of a shallow feature extractor, a deep feature extractor, an upscaling and reconstruction module, and the loss function which dictates

symmetry reinforcement.

Shallow Feature Extractor: This module uses a single convolution layer to reduce spatial size of original EBSD Map and extract shallow features from a given low resolution 4-channel EBSD map (I_{LR}).

$$F_0 = H_{SF}(I_{LR}) \quad (3.1)$$

Here, $H_{SF}(\cdot)$ is a single convolution layer of kernel size 3×3 , which has 4 input channels and 128 output channels. The generated shallow features (F_0) are fed to the deep feature extractor module (H_{DF}).

Deep Feature Extractor: To extract essential deep features from EBSD maps, we implemented deep feature extractors from four well-known single image super-resolution network architectures. These four architectures employ a variety of recent approaches to the SISR problem: deep residual (EDSR) [38], channel attention (RCAN) [39], second order attention (SAN) [40], and holistic attention (HAN) [41] methods. Testing across all four networks enables both a robust analysis of loss functions and a broader understanding of the performance of different architectures in the EBSD-SR problem. It also emphasizes that any deep feature extractor in existence today or developed in the future can be readily applied to this physics-based learning framework.

$$F_{DF} = H_{DF}(F_0) \quad (3.2)$$

Where, $H_{DF}(\cdot)$ is a deep feature extractor module, and F_{DF} is a 128 channels feature map which goes to upscale and reconstruction module.

Upscale and Reconstruction Module: The extracted deep feature (F_{DF}) uses

upscale modules that employ the pixel shuffle operation [58]. The upscaled feature is then mapped into a super-resolved EBSD map using convolution layers that are proportional to the resolution scaling factor.

$$F_{\uparrow} = H_{\uparrow}(F_{DF}) \quad (3.3)$$

$$I_{SR} = H_R(F_{\uparrow}) \quad (3.4)$$

H_{\uparrow} is the upscale module, and H_R is convolution operation of kernel size 3×3 for reconstruction. I_{SR} is the 4 channels generated high-resolution EBSD Map in quaternion space.

3.2.4 Loss Functions

All networks mentioned in Section 3.2.3 were trained using three different loss functions based on a combination of both established practices for the super-resolution problem and the underlying physics associated with EBSD orientation maps. The losses used are traditional L_1 loss and two different physics-guided losses, termed L_1 with symmetry, and approximate rotational distance with symmetry. A histogram comparison of all loss distances for a sample of random orientation vector pairs is shown in Figure 3.7.

L_1 Loss: L_1 loss is a standard norm loss that has been widely used in image restoration tasks and has been shown to have advantages over L_2 loss [59] in terms of sharpness and visual clarity. There is no underlying physical motivation for using L_1 loss beyond its observed advantages in traditional image restoration tasks, which have established the precedent for its use in the SISR problem. The L_1 loss between generated and ground

truth EBSD maps is described using the following equation.

$$L_1 = \frac{1}{N} \sum_{i=1}^N \|H(I_{LR}^i) - I_{HR}^i\|_1 \quad (3.5)$$

Where I_{HR} is the ground truth EBSD map, $H(I_{LR})$ is the generated EBSD map, and N is the batch size.

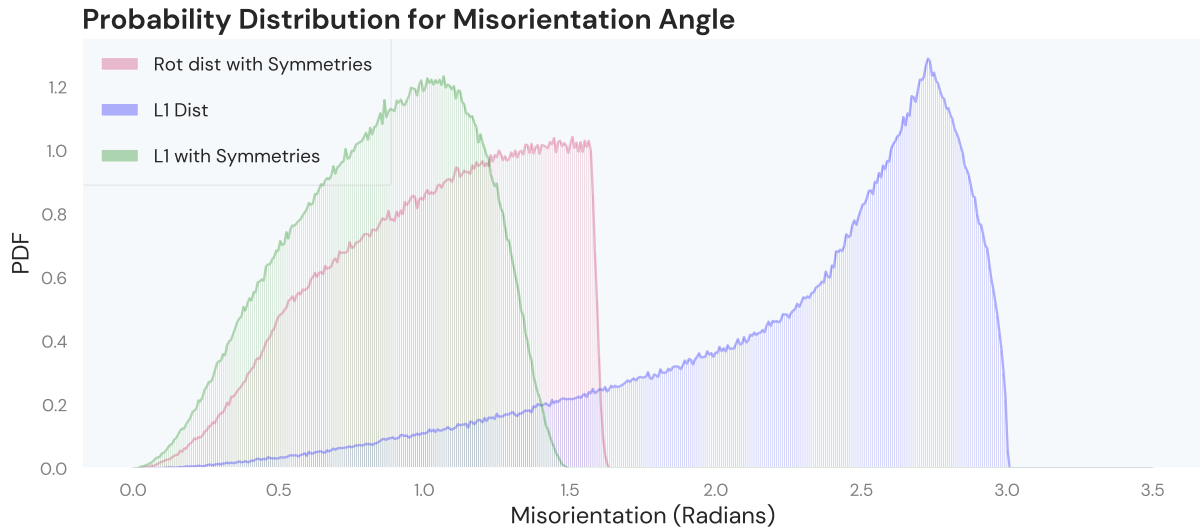


Figure 3.7: ORIENTATION LOSS DISTRIBUTIONS: Probability distribution of loss distances between pairs of randomly sampled 3D rotation vectors. L_1 distance is shown in blue, L_1 distance accounting for crystal symmetry is shown in green, and approximate rotational distance accounting for crystal symmetry is shown in red.

Space Group Symmetry: The orientations in an EBSD map for a given material can only be understood properly in the context of the space group of that material. These same symmetry operators persist in the EBSD diffraction patterns, and create boundaries in orientation space during pattern indexing, dividing the complete sphere of possible quaternion orientations into repeating subsections. For this reason, the crystallographic relationships associated with pixel rotation values in the EBSD maps were accounted for using what we have termed symmetry loss, as shown in Figure 3.6. Symmetries were accounted for according to the space group conventions used to describe crystal symmetry

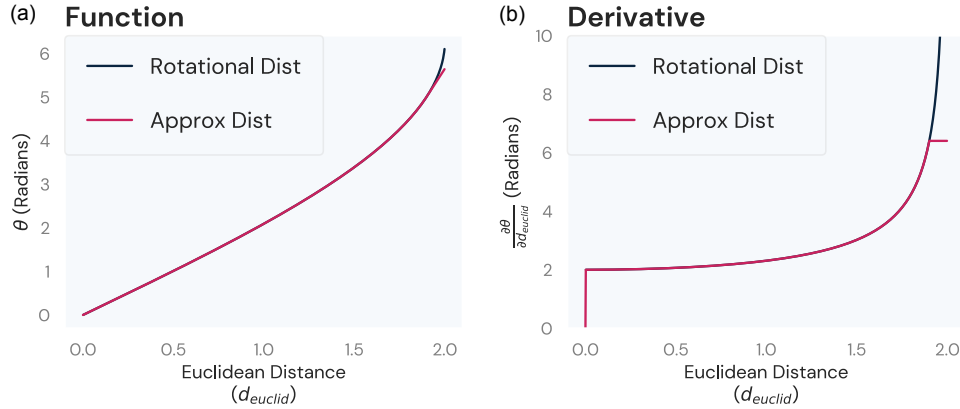


Figure 3.8: APPROXIMATION OF THE ROTATIONAL DISTANCE EQUATION: The derivative is not defined at $d_{euclid} = 2$, so a linear approximation is computed to ensure smooth loss behavior.

systems. This space group information is provided *a priori* during training and inference, but this requirement is not considered overly rigorous, as EBSD measurements typically use *a priori* space group information to simplify the indexing problem. The titanium datasets investigated in this work are part of space group 194, which has a total of 24 symmetries, but only 12 that do not involve a change of handedness. For symmetry-based loss, every pixel value generated by the network is considered as a collection of rotations across all of these symmetries, and the loss distance is calculated as the minimum distance between the ground truth and any value within this collection. For this study, space group symmetry is enforced at the image level. For multi-phase materials, enforcement could also be done at the pixel level through implementation of a phase map.

L_1 Loss with Symmetry: This loss uses L_1 distance to calculate loss magnitude, but incorporates physics to account for space group symmetry in the EBSD map.

Rotational Distance Approximation Loss with Symmetry: Rotational distance loss computes the misorientation angle between the predicted and ground truth EBSD map in the same manner that they would be measured in crystallographic analysis, with approximations to avoid discontinuities. The rotational distance between two

quaternions can be computed as the following:

$$\begin{aligned}
\theta &= 2 \cos^{-1} (\text{Re}(q_1 q_2^*)) \\
&= 2 \cos^{-1} (\langle \vec{q}_1, \vec{q}_2 \rangle) \\
&= 2 \cos^{-1} \left(1 - \frac{1}{2} \|\vec{q}_1 - \vec{q}_2\|_2^2 \right) \\
&= 4 \sin^{-1} \left(\frac{1}{2} \|\vec{q}_1 - \vec{q}_2\|_2 \right) \\
\theta &= 4 \sin^{-1} \left(\frac{d_{\text{euclid}}}{2} \right)
\end{aligned} \tag{3.6}$$

where, $d_{\text{euclid}} = \|\vec{q}_1 - \vec{q}_2\|_2$.

The first order derivative of d_{euclid} is bounded, therefore, d_{euclid} is Lipschitz. However, the gradient of θ goes to ∞ as $d_{\text{euclid}} \rightarrow 2$. To address this issue in training a neural network, a linear approximation was computed at $d_{\text{euclid}} = 1.9$, and utilized for points > 1.9 . This can be seen in Figure 3.8 as a clamp on the max value the derivative of the function can take on. This loss is the most physically accurate of the three considered, and the distribution of loss values for random rotation vectors for rotational distance loss (shown in red in Figure 3.7) matches with the probability distribution of misorientations for hexagonal polycrystals [60].

3.2.5 Inference Pipeline

In order to maximize applicability of EBSD super-resolution to materials research, network output must be interpretable based on established crystallography conventions. To facilitate this, we designed an inference pipeline where network output can be converted to the visualization space used and accepted within the field, as shown in Figure 3.6. During training, we use a physics-based symmetry loss within existing image super-resolution network architectures, but modified to have the appropriate number of input

and output channels. The input to the network is a LR EBSD map, and the generated output is a HR EBSD map, both in the quaternion domain. The physics-based loss, which gives the network information about crystal symmetry, is computed in quaternion orientation space. During inference, the network outputs a HR EBSD map in quaternion space, which is then reduced to the fundamental zone in Rodrigues vector space before being converted into Euler space and projected using inverse pole figure (IPF) color to visualize orientations. IPF color maps are generated with the commercially-available open-source Dream3D software [6].

3.2.6 Network Implementation Details

We use a learning rate of 0.0002, Adam optimizer with $\beta_1 = 0.9$, $\beta_2=0.99$, ReLU activation, batch size of 4 and downscaling factor of 4. The patch size of HR images is 64. The framework is implemented in PyTorch and trained on NVIDIA Tesla V100 GPU. For a batch size of 4 and patch size of 64 across the 3 datasets in this work, the training time for each network for 2000 epochs was approximately 60 hours. Once training is complete, inference time for a given input is on the order of less than one second for an imaging area that would normally take about 10 minutes to gather manually.

In our experimentation with the EBSD super-resolution network, we meticulously tested various batch sizes and patch sizes to optimize network performance. The results, as depicted in the Figures 3.9 and 3.10, clearly indicate that enlarging the patch size enhances the network’s output quality. However, we established a maximum patch size of 64, as we observed negligible improvements beyond this threshold. Conversely, we noted that increasing the batch size adversely affected the network’s output quality. Through these experiments, we determined that a batch size of 4 is optimal for achieving the best results within this deep learning framework.

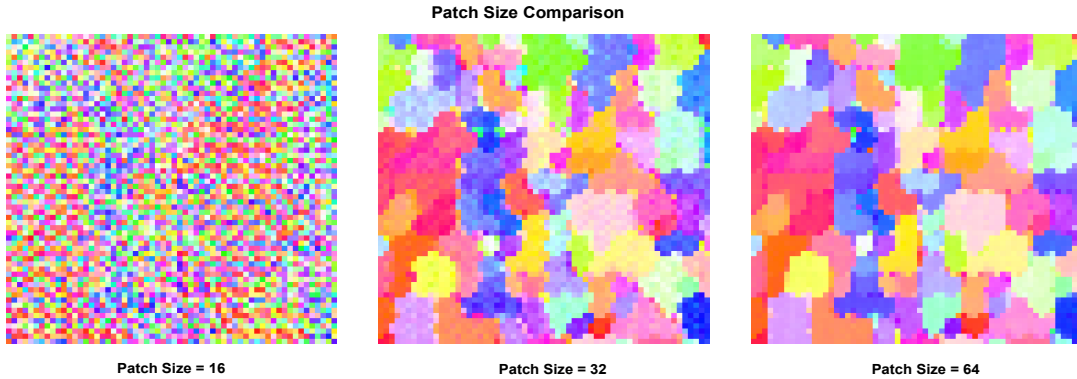


Figure 3.9: NETWORK OUTPUT COMPARISON WITH DIFFERENT PATCH SIZES: Larger patch sizes lead to improvement across all architectures.

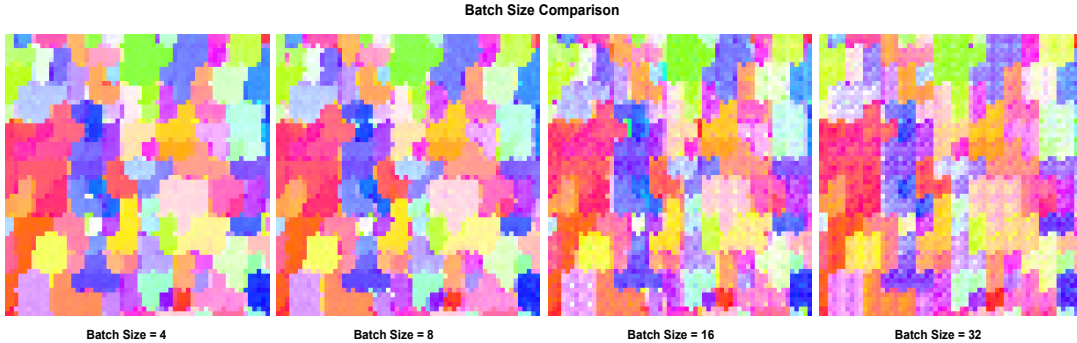


Figure 3.10: NETWORK OUTPUT COMPARISON WITH DIFFERENT BATCH SIZES: Larger batch size leads to lower quality results.

3.2.7 Output Evaluation

For performance evaluation using watershed segmentation, we use a misorientation tolerance of 10 degrees. This is considered a conservative tolerance for feature identification, making it well suited for identification of image artifacts. Watershed segmentation was performed using Dream3D [6]. Network output is evaluated using a combination of image quality and domain relevant metrics. Initially, each set of generated images is evaluated using peak signal-to-noise ratio (PSNR) and structure similarity index measure (SSIM). Images are then segmented into individual grain regions using watershed segmentation based on a relative misorientation tolerance. This approach, coupled with domain knowledge, is commonly used to identify and segment grains in EBSD maps.

3.3 Results

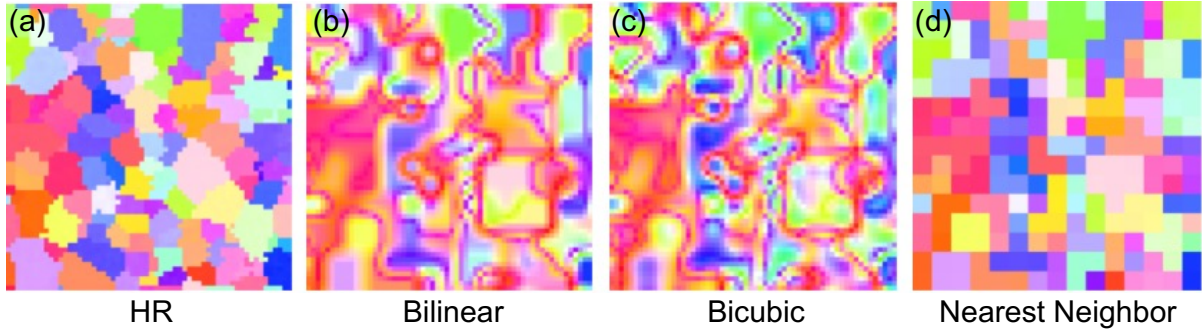


Figure 3.11: COMPARISON OF HR GROUND TRUTH TO TRADITIONAL UPSCALING: Bilinear, bicubic, and nearest neighbor upscaling produce inferior results. Bilinear and bicubic results are nonphysical, and nearest neighbor results are visually identical to LR input.

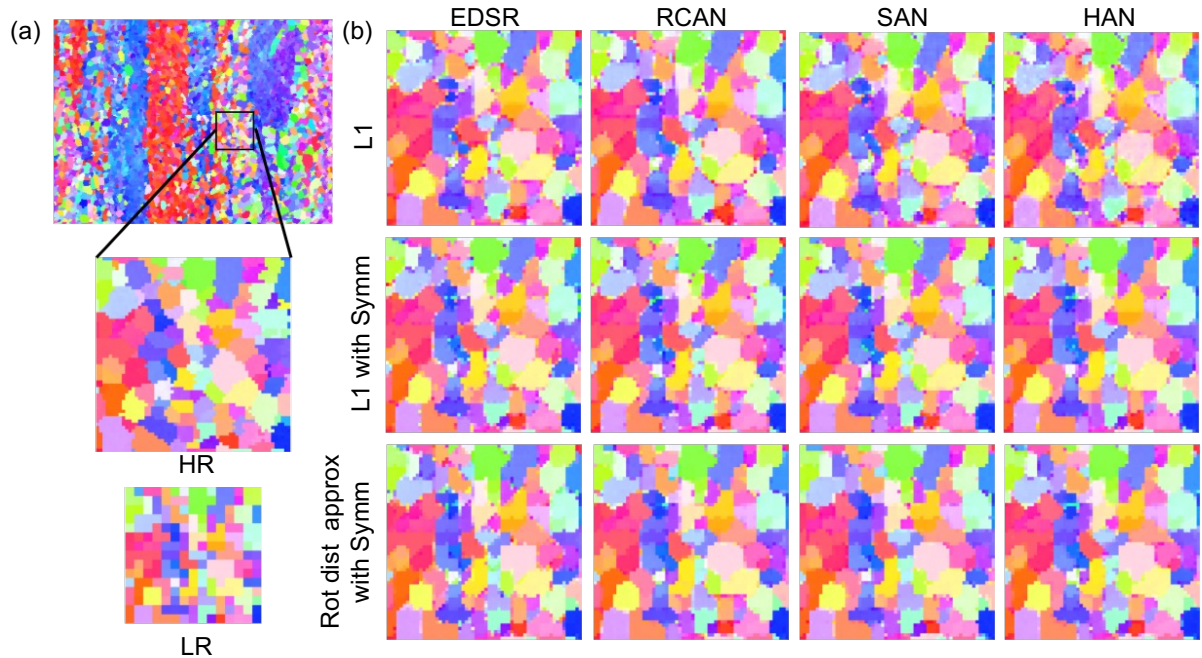


Figure 3.12: VISUAL COMPARISON OF 4X EBSD-SR ON A TEST EBSD MAP: L_1 has non physical structure at grain boundaries, and L_1 with symmetry and approximate rotational with symmetry reduce the non-physical structures at grain boundaries.

3.3.1 Qualitative Output Comparison

To evaluate framework performance, an experimentally gathered dataset of Ti-6Al-4V [1], discussed in Section 3.2.2, was chosen as a candidate dataset for super-resolution testing. This set was of particular interest because of its large number of grains, small grain size, varying local texture, and wide range of represented orientations, all of which make it challenging for the super-resolution problem, as it is desirable to preserve both the gradual orientation gradients within grains as well as the sharp orientation discontinuities at grain boundaries. For training and testing, all LR input was downsampled by a factor of 4 using direct removal of pixel rows and columns to reflect how EBSD resolution would be reduced in actual experiment. When comparing qualitative image results, the reduction from HR to LR by removal of rows and columns from the dataset causes a reduction in visual quality of EBSD maps while offering no possible information inference from subpixel values. This makes shape inference from LR input particularly difficult, as any features removed during downsampling are lost completely. Although it is challenging, our goal is direct application to experimental data, and this form of data loss is exactly what occurs when low-resolution EBSD maps are gathered experimentally. The difficulty associated with this dataset and downsampling is apparent in the traditional bilinear, bicubic, and nearest neighbor upsampling algorithms, shown in Figure 3.11. Bilinear and bicubic approaches produce nonphysical artifacts that do not exist in experimental EBSD maps. These non physical artifacts are the product of interpolations made through quaternion orientation space with no regard for symmetry relationships. Nearest neighbor approximations produce higher resolution visual replicas of the low resolution input.

A full comparison of image quality on the test set for all networks and losses is shown in Figures 3.12, 3.13 and 3.14. All investigated networks produced significantly better results than traditional algorithms, but there were variations in quality across

the different network types and the loss function used. For all networks, grain shapes differed from HR, as many of the fine grain features are lost. This is expected, as the downsampling method removes much of this information, and networks cannot infer on data that does not exist. However, even with different shapes, L_1 and L_1 with symmetry produce grains with smooth contours and a range of facet angles, which is a good reflection of what would be expected in real materials. However, across all networks, both of these losses produce nonphysical image artifacts at the edges of grains. Including physics-based symmetry into L_1 reduces the quantity of these compared to traditional L_1 , but the most dramatic reduction in artifacts occurs for approximate rotational distance with symmetry. Although rotational distance tended to produce slightly more cube-like grain shapes, the overall reduction in visual artifacts makes network output much more physically accurate.

Specifically, we observe that the L_1 loss tends to introduce noise at the grain boundaries, a phenomenon that persists regardless of the network architecture employed. While the integration of symmetry with the L_1 loss mitigates this boundary noise to some extent, it is the rotational distance combined with symmetry that most effectively eliminates this noise across all four network architectures. Consequently, we conclude that the rotational distance with symmetry is the most effective loss function, outperforming the others in enhancing the network’s performance.

3.3.2 Quantitative Output Comparison

Quantitative evaluations of peak signal-to-noise ratio (PSNR) and structural similarity index measure (SSIM) across different architectures and losses are shown in Table 3.1. Comparing between networks, HAN consistently performed best in all cases of physics-based loss. Across all network architectures, the incorporation of physics into the loss

	EDSR	RCAN	SAN	HAN
L ₁ (No Physics)	14.85/ 0.472	14.89 / 0.474	14.83/ 0.465	14.65/ 0.450
L ₁ with Symmetry	15.05/ 0.483	15.07/ 0.484	15.04/ 0.484	15.13 / 0.497
Rot. Dist. Approx with Symmetry	15.02/ 0.486	15.24/ 0.509	15.23/ 0.510	15.30 / 0.513

Table 3.1: **PSNR/SSIM** COMPARISON FOR 4× SUPER-RESOLUTION SCALING: Physics-based loss consistently outperforms Bilinear, Bicubic, Nearest Neighbor, and pure L₁ loss with no physics, regardless of architecture. Rows represent different loss functions and columns represent different network architectures. Higher number is desired for both PSNR/SSIM.

	Bicubic	Bilinear	Nearest Neighbour
Ti-6Al-4V: (PSNR/SSIM)	11.22/ 0.211	11.30/ 0.237	13.25/ 0.373
Ti-7Al 1% (PSNR/SSIM)	18.47/ 0.7135	18.66/ 0.751	22.63/ 0.823
Ti-7Al 3% (PSNR/SSIM)	19.49/ 0.7372	19.65/ 0.773	24.11/ 0.8328

Table 3.2: **PSNR/SSIM** COMPARISON FOR 4× ALGORITHMIC SCALING: PSNR/SSIM values for bilinear, bicubic, and nearest neighbour are consistently lower than network-based methods.

	EDSR	RCAN	SAN	HAN
L ₁ (No Physics)	5.7	5.5	6.0	7.4
L ₁ with Symmetry	4.6	4.9	4.4	2.7
Rot. Dist. Approx with Symmetry	1.8	0.9	1.3	1.4

Table 3.3: **Percentage of Single Pixel Features:** Rows represent different loss functions and columns represent different network architectures. Lower values indicate better performance, with ground truth containing approximately 0.2% single pixel features. Physics based loss reduces noise, leading to lower single-pixel feature counts.

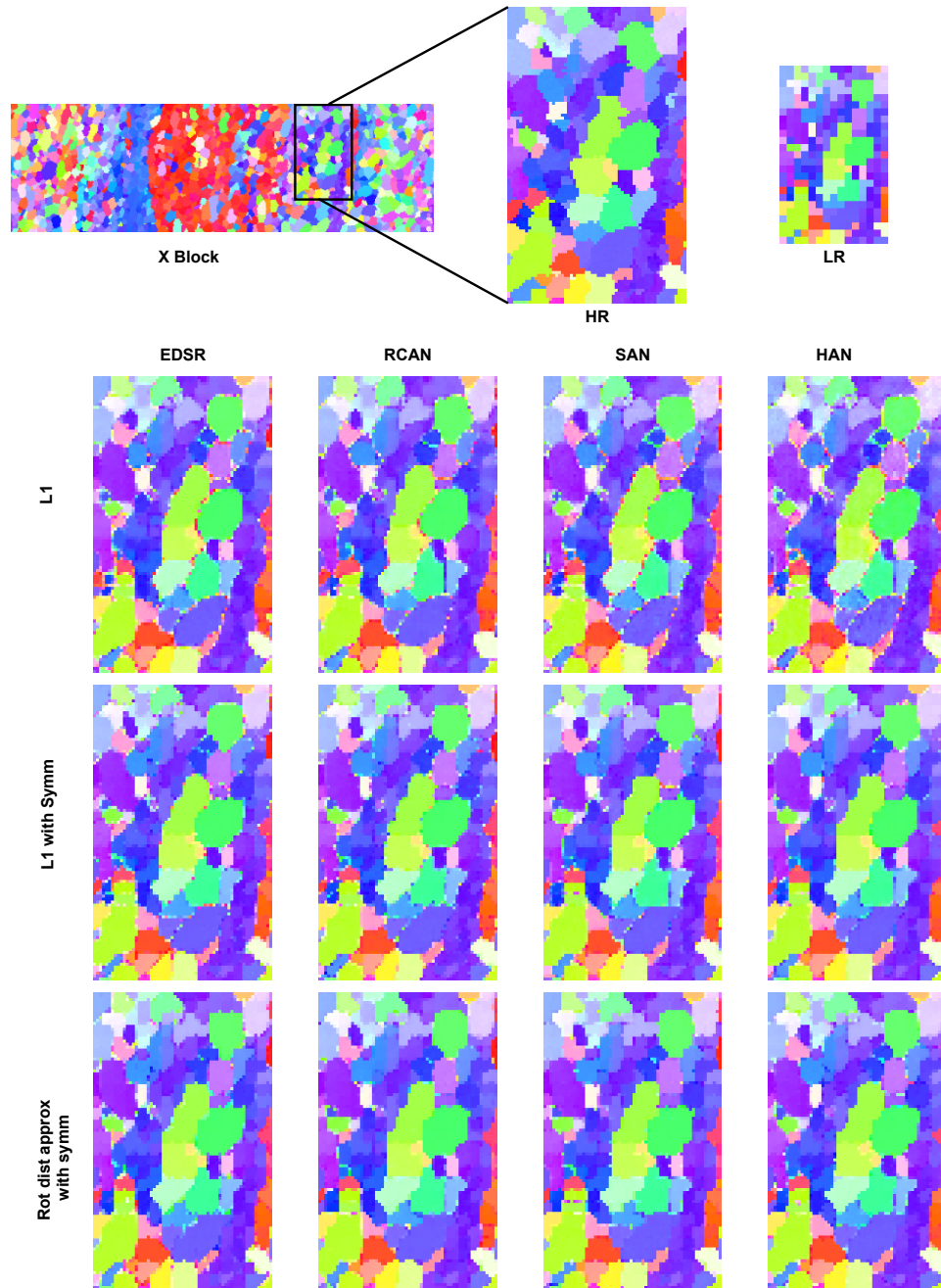


Figure 3.13: VISUAL COMPARISON OF 4X EBSD-SR ON A TEST EBSD MAP: L_1 has non physical structure at grain boundaries, and L_1 with symmetry and approximate rotational with symmetry reduce the non-physical structures at grain boundaries.

function led to improvement in both PSNR and SSIM. Overall, the most physically accurate loss metric, approximate rotational distance, performed best. Bilinear, bicubic,

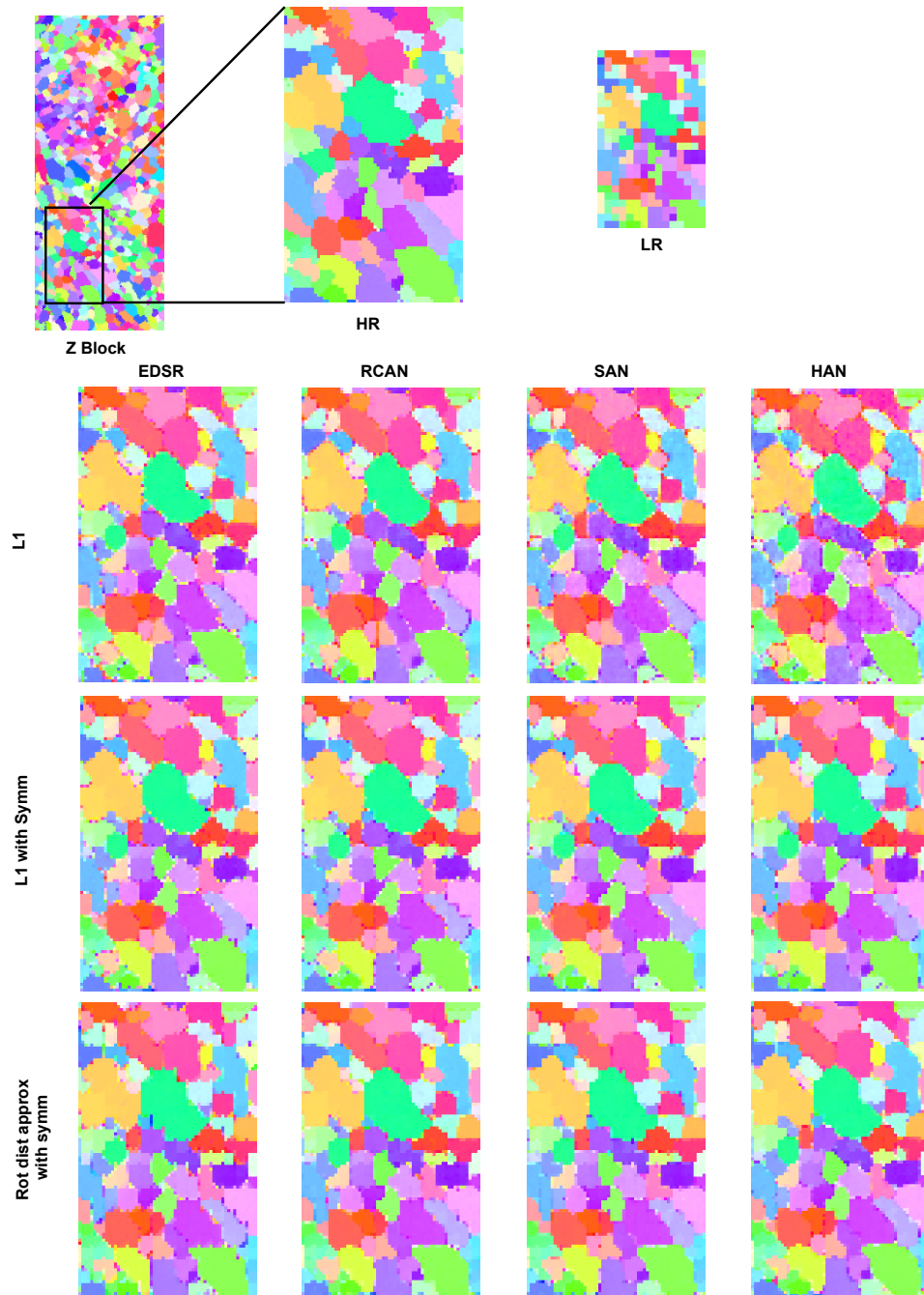


Figure 3.14: VISUAL COMPARISON OF 4X EBSD-SR ON A TEST EBSD MAP: L_1 has non physical structure at grain boundaries, and L_1 with symmetry and approximate rotational with symmetry reduce the non-physical structures at grain boundaries.

and nearest neighbor scores are shown in Table 3.2, but their scores were dramatically lower, and their nonphysicality made further consideration irrelevant.

	EDSR	RCAN	SAN	HAN
L ₁ (No Physics)	522.2	475.7	530.0	628.2
L ₁ with Symmetry	433.3	444.4	415.5	284.1
Rot. Dist. Approx with Symmetry	244.3	152.7	166.4	159.3

Table 3.4: **Percent Feature Difference Over Input:** Rows represent different loss functions and columns represent different network architectures. Lower values indicate better performance. Physics-based loss reduces noise and spurious features, which keeps the overall feature count closer to the amount expected based on input.

Table 3.3 shows the percentage of pixels in the test set identified as single pixel features in a watershed segmentation with a misorientation tolerance of 10° . With high-resolution ground truth having around 0.2% single pixel features, this metric can be considered as a close approximation to the percentage of single-pixel artifacts, which appear as salt-and-pepper noise at grain boundaries. Across all architectures, the incorporation of physics-based loss leads to a clear reduction in single-pixel artifacts. Overall, RCAN with rotational distance loss with symmetry had the fewest single pixel features, but RCAN showed comparatively worse values for L₁ loss with symmetry. These variations in noise are small when compared to the differences between physics-based and non-physics-based loss.

Table 3.4 shows the percentage difference in number of features relative to LR input, when a watershed segmentation with a misorientation tolerance of 10° is applied. A larger percentage value indicates higher feature counts than the LR input, which correlates to occurrence of noise and spurious features, and arise from regions of noise at grain boundaries or overemphasized misorientation gradients within grains. All network outputs produced greater feature quantities than the input, but L₁ loss without symmetry caused the greatest number of spurious features, having over 450% more features than the initial input. Once symmetry is introduced, performance improves across all networks, with HAN producing the fewest excess features for L₁ loss with symmetry, and RCAN

producing the fewest excess features for approximate rotational distance with symmetry.

3.3.3 Effects of Upsampling Layers

Our network architecture is composed of three key components: the Shallow Feature Extractor, the Deep Feature Extractor, and the Upsampling and Reconstruction parts. Notably, in the Upsampling and Reconstruction part, we have implemented a pixelshuffle layer. While it is common practice to employ various other types of upsampling layers, such as bilinear, bicubic, nearest neighbor, and deconvolution, our experiments reveal distinct advantages in using the pixelshuffle layer, particularly for EBSD datasets.

As demonstrated in the accompanying Figures 3.15, 3.16 and 3.17, the pixelshuffle layer emerges as the superior option, markedly outperforming the bicubic and bilinear layers, which tend to generate non-physical results. The nearest neighbor upsampling layer, on the other hand, results in blocky artifacts, whereas the deconvolution layer is prone to producing checkerboard patterns. These findings underscore the effectiveness of the pixelshuffle layer in enhancing the quality of the output while maintaining the physical integrity of the EBSD datasets.

3.3.4 Experiments on Additional Material Datasets

To verify the robustness of the EBSD-SR approach across different materials datasets, the holistic attention network (HAN), which showed consistently strong performance, was trained on two additional experimental datasets of Ti-7Al [2], which were plastically deformed to 1% and 3% strain respectively, and are discussed in Section 3.2.2. These two datasets contain much larger grains than the Ti-6Al-4V set, and also exhibit texture due to plastic deformation. The HAN was trained on all three datasets together rather than on each set individually. Results are shown in Table 3.5, and comparison to traditional

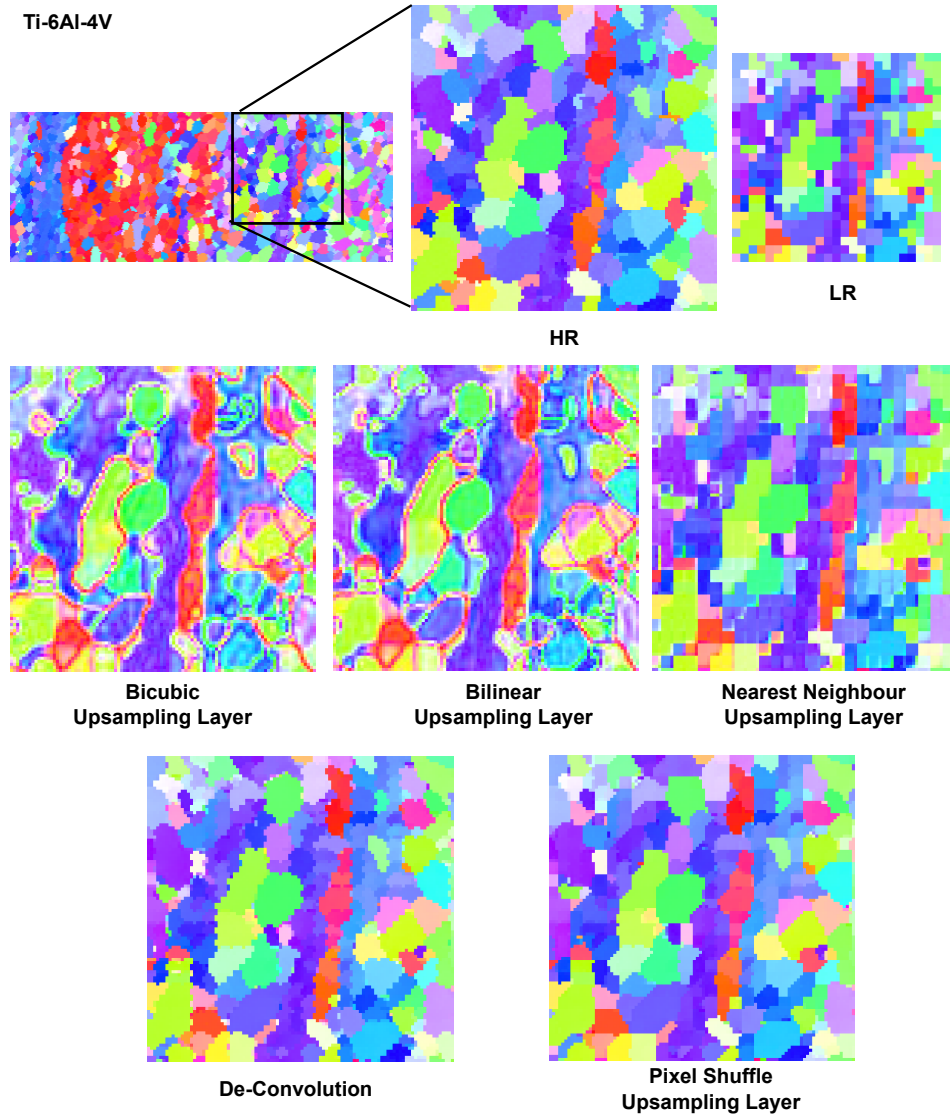


Figure 3.15: COMPARISON OF UPSAMPLING LAYERS FOR Ti-6Al-4V DATASET: Comparison of output of network trained on Ti-6Al-4V with different upsampling layers in upsampling and reconstruction part of network architecture

upsampling algorithms is shown in Table 3.2.

When comparing PSNR/SSIM values, the values for both Ti-7Al datasets are much higher than the those seen for the Ti-6Al-4V set. This is likely due to a combination of grain size and texture differences between sets, with the Ti-6Al-4V set having smaller grains and microtextured regions. Much larger grains makes shape inference at a $4\times$

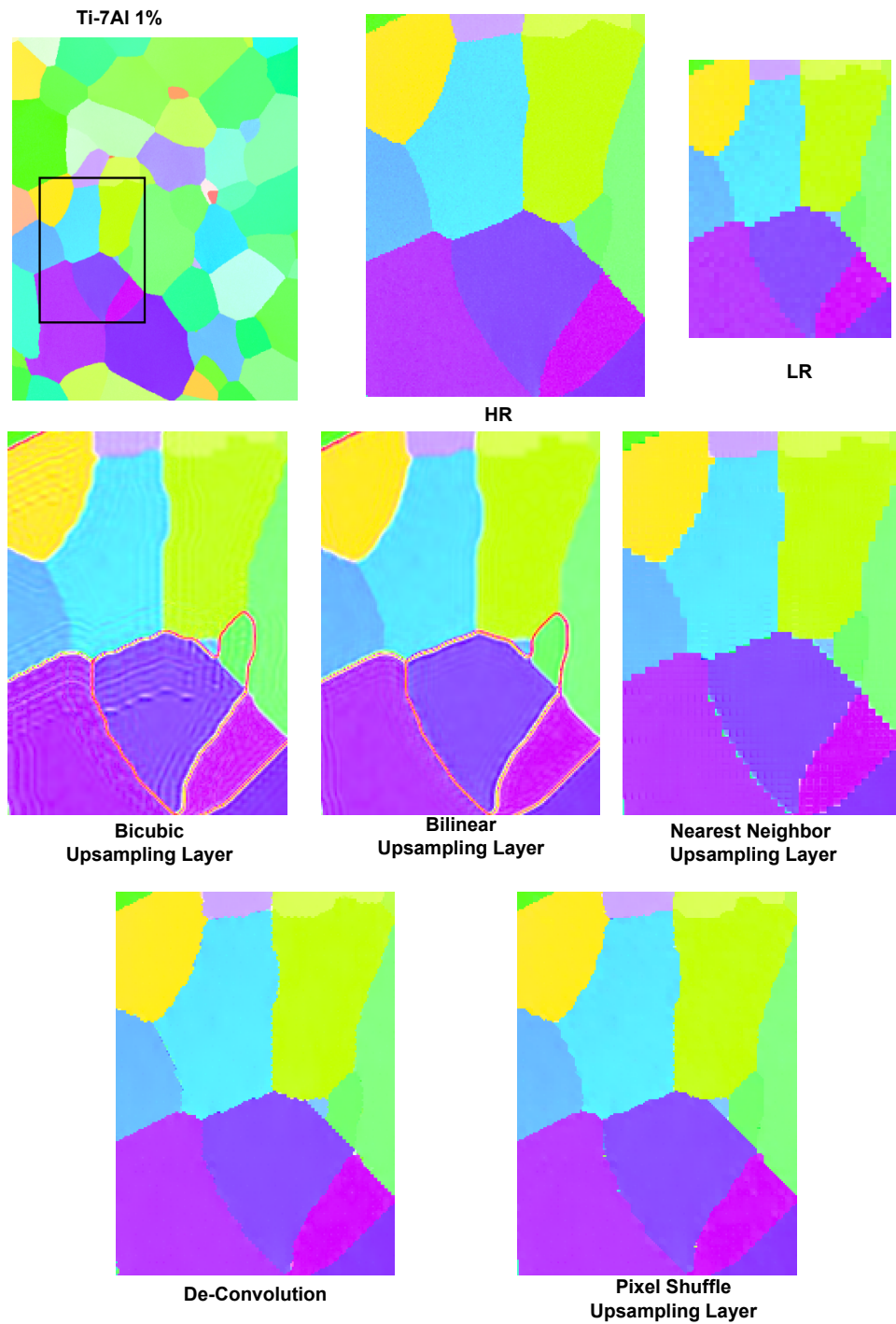


Figure 3.16: COMPARISON OF UPSAMPLING LAYERS FOR Ti-7Al 1 % DATASET: Comparison of output of network trained on Ti-7Al 1 % with different upsampling layers in upsampling and reconstruction part of network architecture

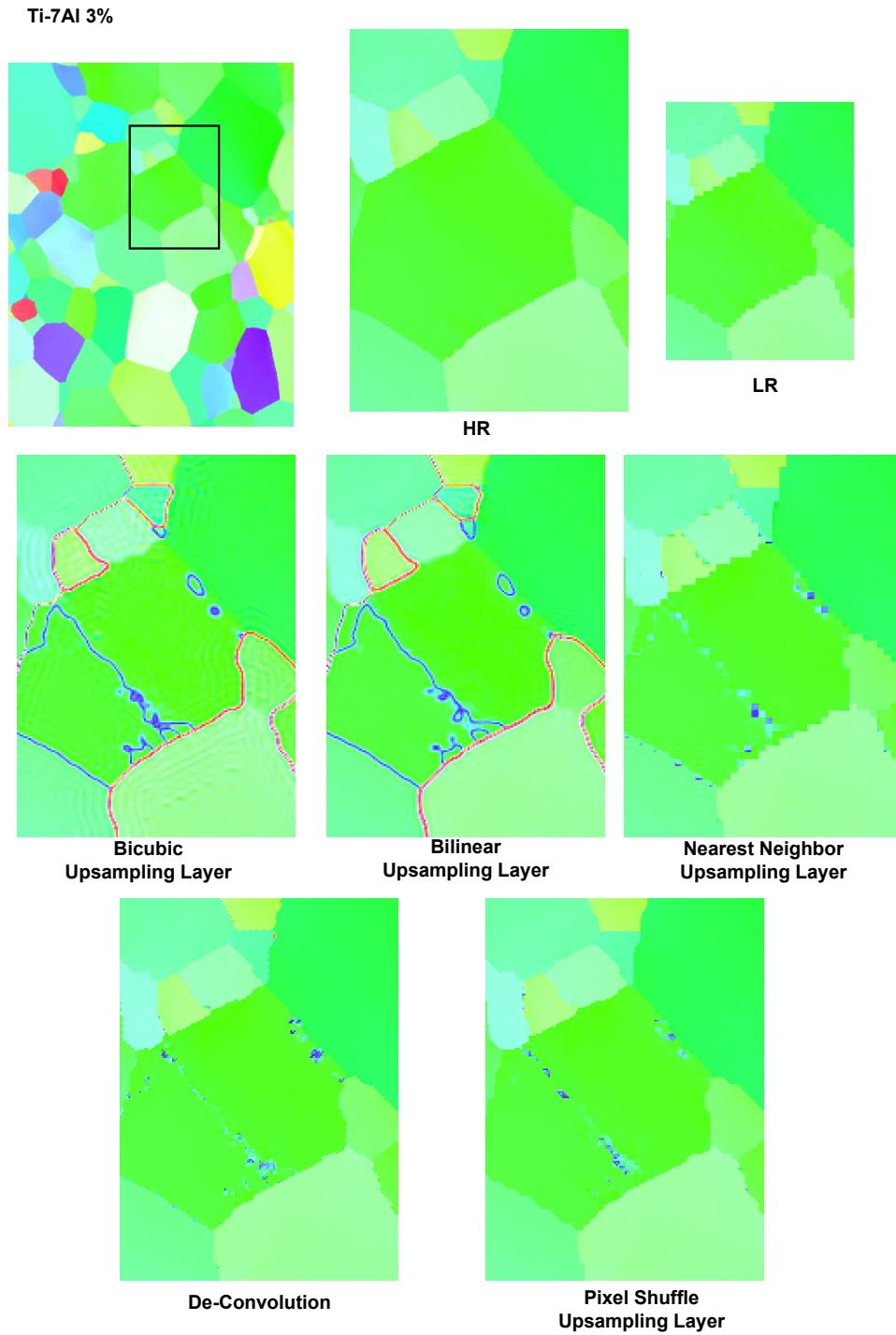


Figure 3.17: COMPARISON OF UPSAMPLING LAYERS FOR TI-7AL 3 % DATASET: Comparison of output of network trained on Ti-7Al 3 % with different upsampling layers in upsampling and reconstruction part of network architecture

	Ti-6Al-4V	Ti-7Al 3%	Ti-7Al 1%
L ₁ (No Physics)	14.94/ 0.478	26.66/ 0.865	25.37/ 0.852
L ₁ with Symmetry	15.19/ 0.497	26.82/ 0.869	25.38/ 0.856
Rot. Dist. Approx with Symmetry	15.35/ 0.5264	27.25/ 0.881	25.72/ 0.871

Table 3.5: **PSNR/SSIM** COMPARISON FOR $4\times$ SUPER-RESOLUTION SCALING ACROSS DIFFERENT MATERIALS: Ti-6Al-4V, Ti-7Al 1%, and Ti-7Al 3% for HAN Network: Physics-based loss consistently outperforms Bilinear, Bicubic, Nearest Neighbor, and pure L₁ loss with no physics, and Columns represent different Titanium datasets. Higher number is desired for both PSNR/SSIM.

scale reduction less difficult and strong material texture makes the range of orientations in the dataset narrower, both of which reduce the burden on the network during the learning/inference process. Regardless of dataset difficulty, rotational distance loss with symmetry consistently produces the highest quality results.

3.4 EBSD SR BisQue Module

We have successfully integrated our Electron Backscatter Diffraction (EBSD) Super-Resolution (SR) inference module into the BisQue platform. On BisQue, users can easily upload a low-resolution EBSD map in numpy format and simply click the 'run' button. Upon activation, the module efficiently processes the input and generates a super-resolved EBSD map, the quality of which is exemplified in the accompanying figure. This user-friendly interface on the BisQue platform makes the application of advanced EBSD SR technology both accessible and convenient. Architecture code is publicly accessible at <https://github.com/UCSB-VRL/EBSD-Superresolution>.

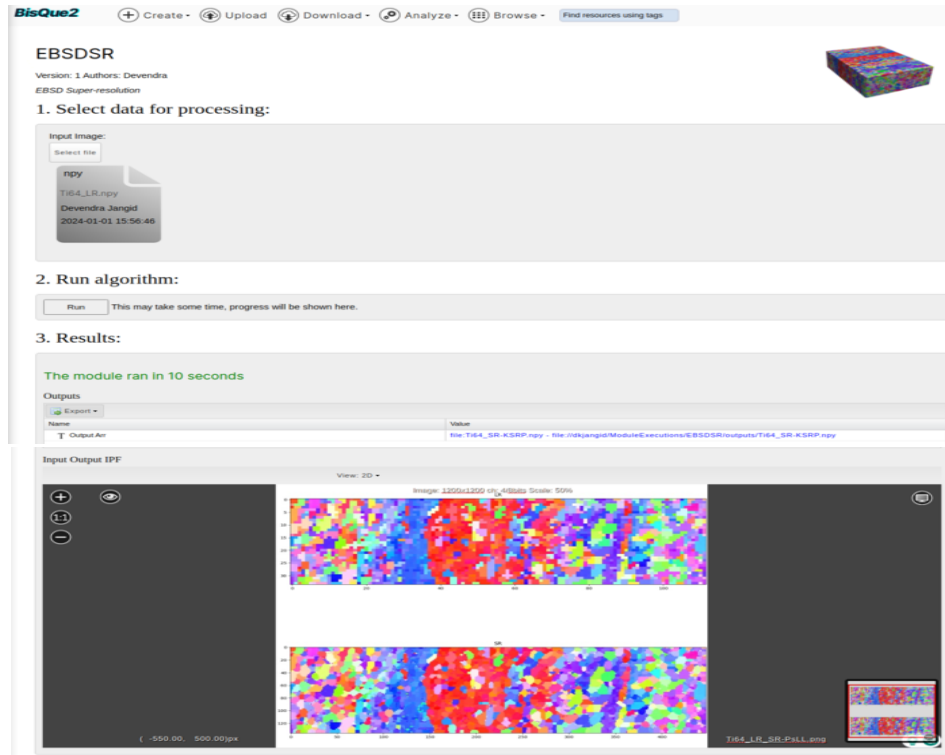


Figure 3.18: BISQUE MODULE FOR EBSD SR: The input for the BisQue model is a low-resolution Electron Backscatter Diffraction (EBSD) map in numpy format. This module efficiently processes the input to generate a high-resolution EBSD map, also in numpy format. Additionally, it facilitates the visualization of the EBSD map through an Inverse Pole Figure (IPF) map, offering a clear and detailed representation.

3.5 Discussion and Conclusion

The results demonstrate that regardless of architecture or approach, the incorporation of domain-related physics into the training process leads to better results for the SISR problem for scientific data. In both qualitative and quantitative evaluation across every metric considered, physics-based loss consistently outperformed traditional L_1 loss regardless of the network used. When comparing physics-based losses specifically, rotational distance consistently outperformed all other losses on all quantitative metrics, though it should be noted that L_1 with symmetry exhibited well-behaved grain shapes in qualitative evaluation while also having quantitative results superior to traditional meth-

ods. For the architectures studied here, comparison across all evaluation metrics shows that attention based models outperform purely residual architectures in the EBSD-SR problem, with holistic attention (HAN) exhibiting the best performance. This result is somewhat expected, as the layer attention module and channel-spatial attention modules present in the HAN architecture provide additional ability to learn spatial correlations across different channels and layers [41]. For this reason, the HAN architecture also exhibited the strongest performance on benchmark datasets for the traditional single image super-resolution problem [41].

The results of this investigation present a strong case for the benefits of an adaptable learning approach that can be applied to future architectures. Although HAN was the best performing network overall, it still was not able to consistently outperform other networks across all metrics presented here. Furthermore, when considering the range of possible microstructures that can exist across 230 different space groups and their multi-phase combinations, it is likely there will never be a single architecture that consistently performs best. The physics-based approach presented here improves performance and presents a path for EBSD super-resolution to keep pace with developments across the SISR field as a whole. This approach can be readily extended to other materials and microstructures using phase masks labeled by space group, accounting for each of their respective symmetries using the methods described here. Going forward, this approach to physics-informed EBSD super-resolution can be used in high-throughput EBSD experiments for the generation of larger, more robust datasets, and also to make higher resolution 3D datasets when combined with asymmetric serial-sectioning approaches (higher resolution in x , and y , lower resolution in z). With spatial super-resolution, the number of time-consuming pattern gathering steps can be reduced. EBSD patterns in 2D, which would normally take minutes or hours to gather can be super-resolved in seconds, and 3D EBSD patterns which would normally take days or weeks to gather can be super-resolved

in minutes. These super-resolution tools can accelerate the materials development process while ensuring that all network learning occurs in the domain boundaries established by physics and crystallography.

In conclusion, we designed an adaptable, physics-guided approach to the super-resolution problem for EBSD orientation maps that employs several deep feature extraction methods from existing single image super-resolution architectures, as well as losses accounting for crystal symmetry and rotation physics. Unlike existing SR methods which operate on scalar image data, the training pipeline is implemented in quaternion orientation space. The inference pipeline produces quaternion output that is converted into Euler angle representation and colored based on IPF projection conventions. Qualitative and quantitative image analysis demonstrate that networks with physics-based learning consistently outperform both traditional upscaling algorithms and analogous network approaches that do not employ physics. Accounting for crystal symmetry in learning leads to increases in PSNR and SSIM, and also reduces single-pixel artifacts and spurious visual features. L_1 loss with symmetry produces well-behaved grain shapes, and approximate rotational distance with symmetry greatly reduces the occurrence of noise and visual artifacts. The presented framework can be readily applied to future super-resolution network architectures.

Chapter 4

Q-RBSA: High-Resolution 3D EBSD Map Generation Using An Efficient Quaternion Transformer Network

In this chapter, we expand our examination of 2D Electron Backscatter Diffraction (EBSD) Super-Resolution (SR) to explore its applications in three dimensions. We introduce a novel neural network architecture, grounded in physical principles, specifically tailored for generating 3D microstructure datasets. The main contribution of this chapter lies in the design of a quaternion-based neural network architecture and the development of a comprehensive pipeline for the synthesis of 3D microstructure datasets.

Gathering 3D material microstructural information is time-consuming, expensive, and energy-intensive. Acquisition of 3D data has been accelerated by developments in serial sectioning instrument capabilities; however, for crystallographic information, the electron backscatter diffraction (EBSD) imaging modality remains rate limiting. We propose a physics-based efficient deep learning framework to reduce the time and cost of collecting 3D EBSD maps. Our framework uses a quaternion residual block self-

attention network (QRBSA) to generate high-resolution 3D EBSD maps from sparsely sectioned EBSD maps. In QRBSA, quaternion-valued convolution effectively learns local relations in orientation space, while self-attention in the quaternion domain captures long-range correlations. We apply our framework to 3D data collected from commercially relevant titanium alloys, showing both qualitatively and quantitatively that our method can predict missing samples (EBSD information between sparsely sectioned mapping points) as compared to high-resolution ground truth 3D EBSD maps. Compared to the traditional signal processing method which uses local neighborhoods, our method uses adaptive receptive fields of different spatial sizes with self-attention layers to learn both local and long-range correlations for material datasets: Ti-6Al-4V, and Ti-7Al. The contents of this chapter are discussed in our published paper [8].

4.1 Introduction

In the pursuit of materials development for extreme environments to understand the limit of its performance, 3D microstructural information is essential input for structure-property models [16]. Many engineering materials are polycrystalline, meaning they are composed of many smaller crystals called grains, and the arrangement of these grains impacts their thermomechanical properties. To collect crystallographic microstructure information, 3D microscopy techniques have been developed that span lengthscales from nanoscale to mesoscale [61]. These experiments require costly or challenging to access equipment, like synchrotron light sources for high X-ray fluxes [62–64], precise automated robotic mechanical polishing and imaging [65, 66], or high-energy ion beams and/or short pulse lasers coupled to electron microscopes [32, 67].

Recent advances in 3D characterization have reduced the time required for data collection, but serial sectioning methods (where material is progressively removed from the

sample between images) are still slow processes that require expensive microscopes [23]. During serial sectioning for microstructural information, typical experimental steps might include material removal and cleanup (mechanical polishing, laser ablation, focused ion beam milling) and imaging for orientation or chemical information. As such, any efforts that reduce the total number of required serial sections in a 3D experiment will ultimately lead to substantive time and cost savings. This cost motivates methods to reduce the number of required points, such as smart or sparse sampling [68–70], or machine learning super-resolution [35]. In these methods, missing information can be inferred using interpolation-based algorithms (bicubic, bilinear, or nearest neighbor) or data-based learning. Recent progress in computer vision [35, 40, 71] has shown that the generation of missing samples/data with data-based learning outperforms traditional interpolation algorithms for RGB images. However, unlike RGB images, EBSD maps carry embedded crystallography, so existing learning-based methods are not well suited to generate missing EBSD data.

In chapter 3, we developed a deep learning framework for 2D super-resolution that utilized an orientationally-aware physics-based loss function to generate high-resolution (HR) EBSD maps from experimentally gathered low-resolution (LR) maps. This approach allowed for significant gains in 2D resolution, but expansion to 3D remained difficult due to data availability limitations as 3D EBSD is expensive and time consuming to gather. To address this, here we propose a 3D deep learning framework based on quaternion convolution neural networks with self-attention alongside physics-based loss to super-resolve high resolution 3D maps using as little data as possible. Using real-valued convolution for quaternion-based data has been shown to be inefficient in learning the inter-channel relationship (see equation 2.1) that arise from quaternion vector components (i, j, k component) [28]; which leads to longer training times and larger data burdens. We demonstrate that a quaternion-valued neural network is more efficient and

produces better results than real-valued convolution neural networks such as those used in previous work [72].

The crystallographic information contained in EBSD maps is generally expressed in the form of crystal orientations spatially resolved at each pixel or voxel. These orientations, like other rotational data, can be expressed unambiguously using quaternions. They therefore can be incorporated into network architecture as prior information by using quaternion-valued convolution for local-level correlation, rather than real-valued or complex-valued convolution. The basic component in traditional CNN-based architectures is real-valued convolutional layers, which extract high-dimensional structural information using a set of convolution kernels. This approach is well-suited for unconstrained image data like RGB, but when convolution kernels fail to account for strict inter-channel dependencies where present, the result is greater learning complexity. Some successful efforts have been made to design lower-complexity architectures by extending real-valued convolution to complex-valued convolution [73,74] and quaternion-valued convolution [75–77] in the field of robotics [78], speech and text processing [28], computer graphics [79,80], and computer vision [77,81]. Although these convolution layers are useful to learn local correlations, they struggle to learn long-range correlations. In contrast, transformer-based architectures have recently shown significant success in learning long-range correlations in natural language [82] and vision tasks [71,83]. However, the computational complexity of transformer-based architectures grows quadratically with the spatial resolution of input images due to self-attention layers, so transformers alone are not well-suited for restoration tasks. However, recent work by Zamir [71] proposed self-attention across channel dimensions to reduce complexity from quadratic to linear with progressive learning for image restorations and showed superior results to convolution-based architecture alone.

Inspired by this idea, we propose the use of quaternion self-attention for EBSD

super-resolution, using physics-aware quaternion convolution for orientation recognition, a physics-based loss function that is sensitive to material crystal symmetry, and progressive learning to incorporate long-range material relationships. Physics-aware quaternion convolution follows the approach of [28, 73, 84], where convolution is depth-wise across channel dimension and uses a reduced number of interdependent weights whose connectivity is based on the Hamiltonian product, as explained in equation 4.2. We use a loss function that accurately measures the crystal orientations in EBSD maps and also accounts for the hexagonal close-packed symmetry present in α -phase Ti-6Al-4V and Ti-7Al, the two alloys investigated here. Finally, progressive learning refers to having variable patch sizes instead of fixed patch sizes during training, which is relevant for most engineered material microstructures, where important features can span across length scales (and patch sizes). The titanium alloys studied herein are well-known to have many different microstructural variants accessible via processing, resulting in varying grain size and morphology. For the datasets that we consider specifically, the Ti-6Al-4V variant has smaller equiaxed grains, while the Ti-7Al alloy has much larger grain size, so applying a fixed patch size across these two materials would be sub-optimal. To enforce long-range learning among these grain features, we used progressive patch sizes starting from 16 to 100 during the training of the network. Training behaves in a similar fashion to curriculum learning processes where the network starts with a simpler task and gradually moves to learning more complex ones.

4.2 Method

In this section, we discuss about the datasets, network architecture, loss function, and training strategy of neural network.

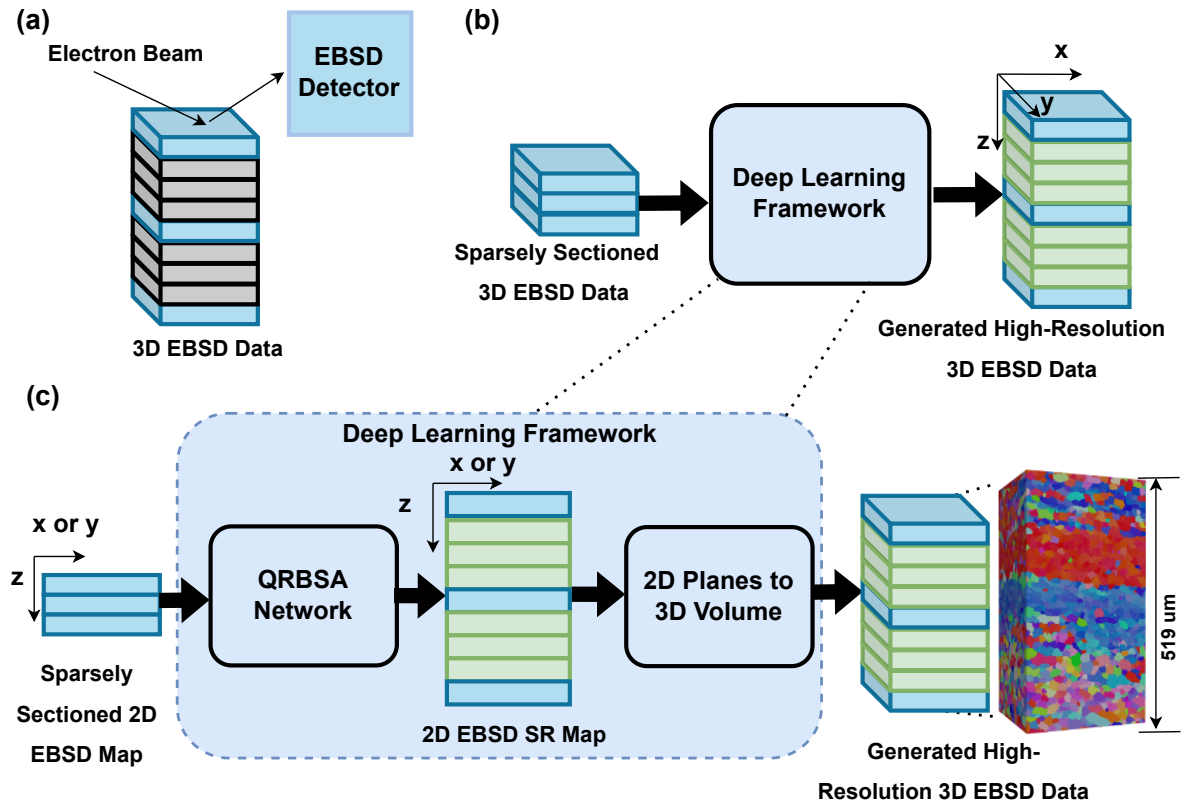


Figure 4.1: Q-RBSA EBSD RESOLUTION ENHANCEMENT FRAMEWORK: In the experimental pipeline shown in (a), material researchers collect EBSD orientation information for each (x,y) coordinate in a given sectioning plane, and then remove material using laser ablation or robotic polishing to reach the next plane in the z direction to build a 3D volume. In our framework (b), researchers collect EBSD information from a reduced set of points (blue planes), omitting some planes that would normally be gathered (gray planes). The missing information (green planes) are then generated in 2D as a series of (x,z) or (y,z) planes by our quaternion-based, physics-informed deep learning framework, shown in (c). Here, the network takes advantage of orthogonal independence to efficiently generate 3D volumes using less data, as large amounts of EBSD are costly and the choice of serial sectioning direction has minimal impact on the resultant final volume.

4.2.1 Dataset Preprocessing

High-Resolution (HR) Ground Truth: The ground truth data is experimental 3D EBSD data gathered from the titanium alloys, Ti-6Al-4V and Ti-7Al (one Ti-7Al sample deformed in tension to 1% and one to 3%), using a commercially-available rapid-serial-sectioning electron microscope known as the Tribeam [32, 33]. The Ti-6Al-4V dataset, shown in Figure 4.2(a), is of pixel size $346 \times 142 \times 471 \times 4$ ($z \times y \times x \times ch$), where the last dimension is the quaternion component. Analogously, the Ti-7Al shown in 4.2(b) and 4.2(c) are of size $232 \times 674 \times 770 \times 4$ ($z \times y \times x \times ch$) and $224 \times 770 \times 770 \times 4$ ($z \times y \times x \times ch$) pixels respectively, with all edges cropped to produce a perfect parallelepiped volume. Each voxel in the Ti-6Al-4V set has resolution of $1.5 \times 1.5 \times 1.5 \mu\text{m}$, and in both Ti-7Al sets, each voxel has a resolution of $1.3 \times 1.3 \times 1.3 \mu\text{m}$.

These titanium alloys are composed primarily of the hexagonal close packed grains. The EBSD ground truth data was indexed using a combination of EMsoft dictionary indexing (Ti-7Al 1% and 3%) [20] or spherical indexing (Ti-6AL-4V) [21] for improved indexing accuracy beyond conventional Hough transform indexing. Experimental data was cleaned using a minimum size filter of 27 voxels in volume and a minimum feature neighbor filter of 2 neighbors per grain (applied using DREAM.3D [6]) to eliminate grains that were insufficiently or inaccurately resolved by the 3D characterization technique.

In total, the Ti-6Al-4V dataset contains about 57,000 grains, visible in the IPF maps as regions of different color. The Ti-7Al material has larger grain size, with 500-1000 grains in each dataset. The datasets were proportionally divided into training, validation, and test subsets in ratios of 65%, 15%, and 20% respectively.

Sparsely Sectioned Input EBSD Data: Sparsely sectioned EBSD data are down-scaled versions of the high-resolution EBSD data. However, because of how EBSD information is gathered, these EBSD data are not downscaled using blurring. Instead

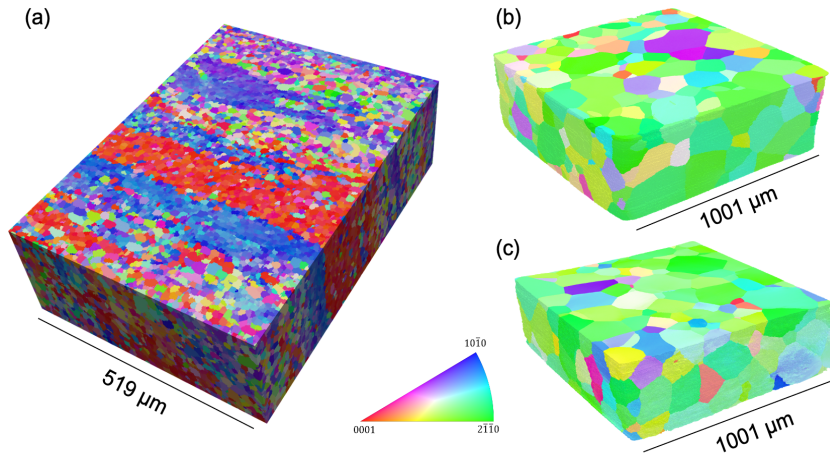


Figure 4.2: RENDERING OF 3D EBSD DATASET INVESTIGATED: shown in IPF coloring of the titanium alloys, (a) Ti-6Al-4V and (b) Ti-7Al mechanically loaded in tension to 1% strain and (c) 3%, used for network training, testing and validation. Dataset details are available elsewhere [1, 2].

sparsely sectioned EBSD data are produced by removal of xy planes in the z direction from the high-resolution ground truth with a downscale factor of 4x and 2x ($LR = \frac{1}{4}HR$ or $LR = \frac{1}{2}HR$). This is done to imitate the skipping of collection planes that would occur in a 3D experiment with more sparsely sectioned EBSD data (i.e. thicker section depth), which would not influence the electron beam-material but rather lead to larger amounts of material removal between consecutive planes.

4.2.2 Deep Learning for EBSD

The objective is to generate missing sample planes from experimental 3D EBSD data that is sparse along the z -axis. In this approach, material researchers collect sparsely sectioned 3D EBSD data (blue planes) as shown in Figure 4.1 (a), due to the high cost associated with serial sectioning and collecting 3D EBSD data at higher resolution. Ideally, a 3D deep learning framework would be designed to generate the missing planes (gray planes), but experimental EBSD data is costly to gather, so available 3D data for training is extremely limited. Additionally, 3D neural networks require more learned

parameters, which, with limited available data, increases the likelihood of overfitting. Instead of a full 3D architecture, a deep learning network is implemented on 2D EBSD maps orthogonal to the sectioned planes, shown as the xz or yz planes in Figure 4.1 (c). Our network takes sparsely sectioned xz or yz EBSD maps as input to generate the missing rows normal to the z -axis. The generated 2D EBSD maps are then combined into a 3D volume. EBSD collection is a point-based scanning method that is directionally independent; therefore missing z rows can be generated from xz or yz EBSD maps, and two 3D volumes can be formed from each sparsely sectioned dataset.

4.2.3 Network Architecture

Although EBSD maps are visualized similarly to RGB images, they are multidimensional maps with inter-channel relationships, where crystal orientation is described using Euler angles, quaternions, matrices, or axis-angle pairs. In chapter 3, we have demonstrated that quaternion EBSD representation is well-suited to orientation expression for loss function design, due to its efficient rotation simplification and avoidance of ambiguous representation. However, we previously used real-valued convolution layers to learn features, which is sub-optimal for EBSD orientation maps where orientations are represented as unit-vector quaternion rotations. Generally speaking, convolution networks provide local connectivity and translation equivariance, which are desirable properties for images, but if additional feature correlations are going to be learned efficiently, it is critical to encode relevant structural modalities into the network architecture and loss function. Real-valued convolution can still learn unit quaternion inter-channel information, but it requires extra network complexity, and by consequence, additional data to inform that complexity. Here, the use of quaternion convolution efficiently encodes prior orientation information into kernels, and also has the advantage of reducing the number

of trainable parameters by 4, as explained the appendix.

4.2.4 Quaternion Convolution

Quaternion convolution neural networks (QCNN) [85] use basic quaternion convolution operation which computes the Hamilton product between the input feature maps and kernel filters rather than just computing correlations between them, as is done in real-valued convolution [86]. For instance, if we consider P_{input} as an input feature map of size $(4K_2 \times H \times W)$, and F as a quaternion kernel filter of size $(4K_2 \times f \times f)$, where K_2 is the number of kernel filters in the previous layer, H and W are the height and width of the input feature map (P_{input}), and f is the spatial size of the quaternion kernel filter (F), then we can split the input feature map (P_{input}) into four components (P_R, P_X, P_Y, P_Z) along the channel dimension, where each component has a dimension of $(K_2 \times H \times W)$. Similarly, the kernel filter (F) can be divided into four components (F_R, F_X, F_Y, F_Z) along the channel dimension, where each component has a dimension of $(K_2 \times f \times f)$. The quaternion convolution ($QConv$) of input feature maps (P_{input}) with a single kernel filter (F) is defined as follows

$$P'_{quaternion} = F \otimes P_{input} \tag{4.1}$$

$$\begin{bmatrix} P'_R \\ P'_X \\ P'_Y \\ P'_Z \end{bmatrix} = \begin{bmatrix} F_R * P_R - F_X * P_X - F_Y * P_Y - F_Z * P_Z \\ F_X * P_R + F_R * P_X - F_Z * P_Y + F_Y * P_Z \\ F_Y * P_R + F_Z * P_X + F_R * P_Y - F_X * P_Z \\ F_Z * P_R - F_Y * P_X + F_X * P_Y + F_R * P_Z \end{bmatrix} \tag{4.2}$$

Here, \otimes is the Hamilton product, and $*$ represents real-valued convolution operation [86]. The output quaternion feature map ($P'_{quaternion}$) has a dimension of $(4 \times H \times W)$ for a single kernel filter, where each component (P'_R, P'_X, P'_Y, P'_Z) has a dimension of

$(1 \times H \times W)$. H and W are the height and width of the output feature maps. For a better understanding of quaternion convolution, please refer to the Figures 4.14, 4.15 and 4.16 provided in appendix section.

4.2.5 Feature Extractors

Our network architecture shown in Figure 4.3 consists of two feature extractors components: a shallow feature extraction module and a deep feature extraction module as discussed below.

Shallow Feature Extractor: This module uses a single quaternion convolution layer ($QConv$), explained in equation 4.2, to reduce the spatial size of sparsely sectioned EBSD maps, while extracting shallow features.

$$F_0 = H_{SF}(I_{LR}) \quad (4.3)$$

Here, I_{LR} is a sparsely sectioned 2D EBSD map and $H_{SF}(\cdot)$ is a single quaternion convolution layer of kernel filter size 3×3 , which has 4 input channels and 128 output channels. The generated shallow features (F_0) are given to the deep feature extractor module (H_{DF}).

Deep Feature Extractor: To learn from sparsely sectioned EBSD maps, our deep feature extractor module uses stacked quaternion residual self-attention (QRSA) blocks to extract high-frequency information and long skip connection to bypass low-frequency information. Residual blocks allow for a deeper network architecture, which provides a larger receptive field and better training stability. In our QRSA module, we use both CNN and transformer ideas to combine the effectiveness of the locality of CNNs with the expressivity of transformers that enables them to synthesize high-resolution EBSD maps. The CNN structure offers local connectivity and translation equivariance, allow-

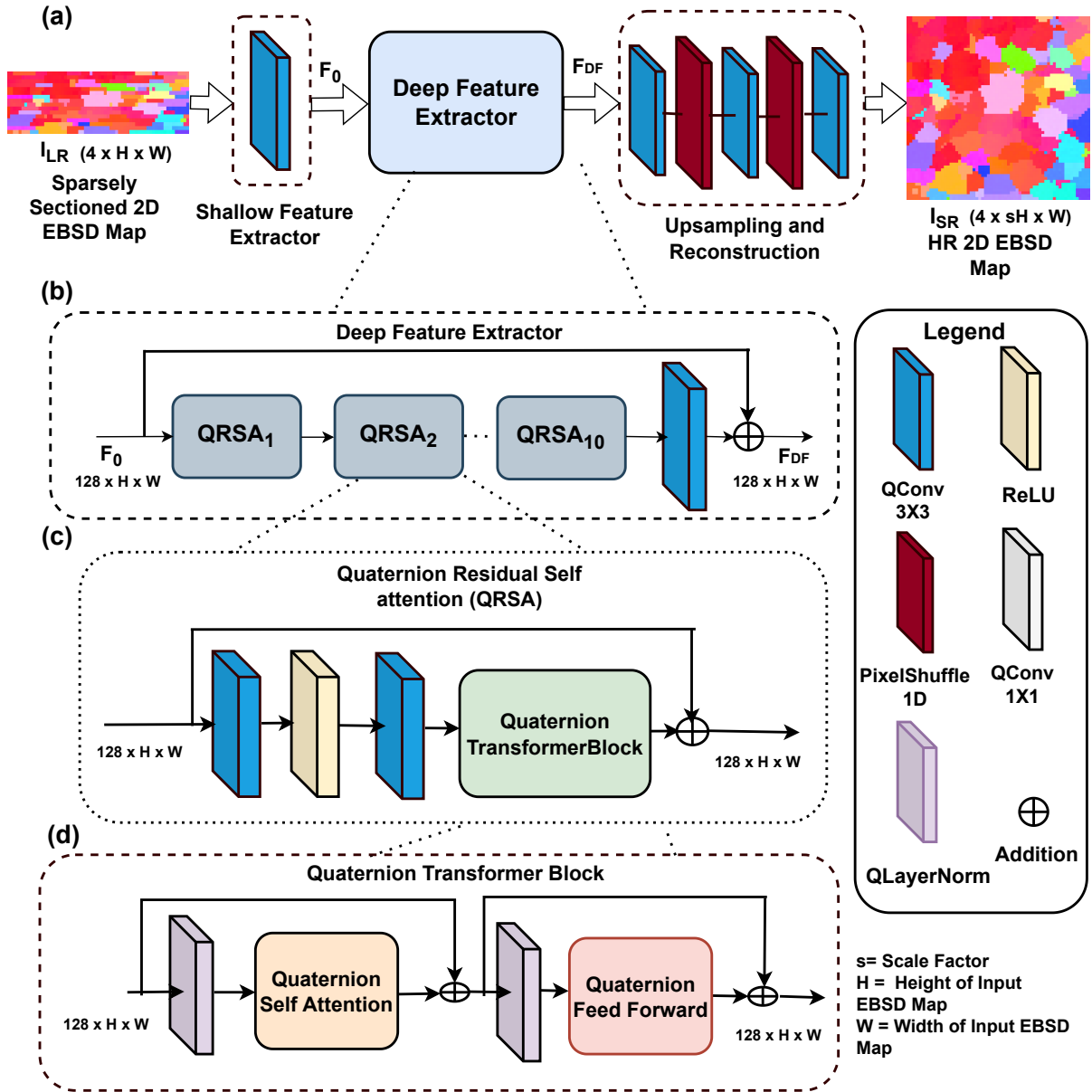


Figure 4.3: QUATERNION RESIDUAL BLOCK SELF-ATTENTION (QRBSA) NETWORK: A sparsely sectioned 2D EBSD map is given to the QRBSA network (a) to generate a high-resolution 2D EBSD map. QRBSA consists of three parts: a Shallow feature extractor, a Deep feature extractor, and Upsampling and Reconstruction. The deep feature extractor uses a residual architecture (b) where residual self-attention blocks (c) are modified with quaternion convolution layers and transformer blocks (d) to efficiently handle orientation data. Quaternion convolution is used to learn local-level relationships, while quaternion transformer blocks learn the global statistics of feature maps. Pixelshuffle layer, modified for 1-dimensional upsampling, is used in the upsampling and reconstruction block to upsample feature maps.

ing transformer components to freely learn complex and long-range relationships. Each quaternion residual self-attention (QRSA) block consists of two quaternion convolution layers and a piece-wise ReLU activation, explained in equation 4.11, between them, and a quaternion transformer block. The quaternion convolution layers with piece-wise ReLU activation help in learning the local structure of extracted shallow features, while the quaternion transformer block captures long-range correlations among features. The short-skip connection is useful to bypass low-frequency information during training.

$$F_{DF} = H_{DF}(F_0) \tag{4.4}$$

$$H_{DF} = QRSA_1 \circ QRSA_2 \circ \dots \circ QRSA_i \dots \circ QRSA_{10} \circ QConv + I \tag{4.5}$$

where, $H_{DF}(\cdot)$ is a deep feature extractor module, and F_{DF} is a 128 channels feature map which goes to the upscale and reconstruction module. $QConv$ is a quaternion convolution layer as explained in equation 4.2, $QRSA_i$ is a i^{th} quaternion residual self-attention (QRSA) block, and I is an identity feature maps.

4.2.6 Quaternion Transformer Block

The standard transformer architecture [82] consists of self-attention layers, feedforward networks, and layer normalization. The original transformer architectures [82,87] are not suitable for restoration tasks due to the requirement of quadratic complexity of spatial size $\mathcal{O}(W^2H^2)$, where W, H is the spatial size of images or EBSD maps. Similar to the approach of [71] and as shown schematically in Figure 4.4, we compute attention maps across the features dimension, which reduces the problem to linear complexity. However instead of depthwise convolution, we use quaternion convolution as explained in equation 4.2, which can be considered as a combination of depthwise convolution and group convolution, but with four-dimensional quaternion constraints. We have also incor-

porated an equivalent quaternion-based gating mechanism into the feedforward network within the transformer, and the traditional convolution used in [71] has been replaced with quaternion convolution layers as explained in equation 4.2 to account for EBSD data modalities. Layer normalization plays a crucial role in the stability of training in transformer architectures. The quaternion layer-normalization is equivalent to the real-valued one by computing normalized features across each component of the quaternion separately, and allows the building of deeper architectures by normalizing the output at each layer. From the normalized features, the quaternion self-attention layer first generates query (Q), key (K) and value (V) projections enriched with the local context. After reshaping query and key projection to reshaped query (Q_r) and reshaped key (K_r), a transposed attention map (A) is generated. The refined feature map, which has global statistical information, is calculated from the dot product of the value projection (V_r) and the attention map (A).

$$\text{Transposed Attention Map (A)} = \text{Softmax}(K_r \cdot \frac{Q_r}{\alpha}) \tag{4.6}$$

$$\text{Quat-SelfAttention}(Q_r, K_r, V_r) = V_r \cdot A \tag{4.7}$$

$$\text{Gating}(X_2) = \text{GeLU}(W_1(X_2)) \odot W_2(X_2) \tag{4.8}$$

Where \odot represents elementwise multiplication, α is a learnable scaling parameter to control the magnitude of the dot product of K_r and Q_r before applying softmax function and GeLU is Gaussian Error Linear Units activation function [88], and W_i ($i=1,2$) is a combination of quaternion convolution layers with kernel size 1 and 3, respectively.

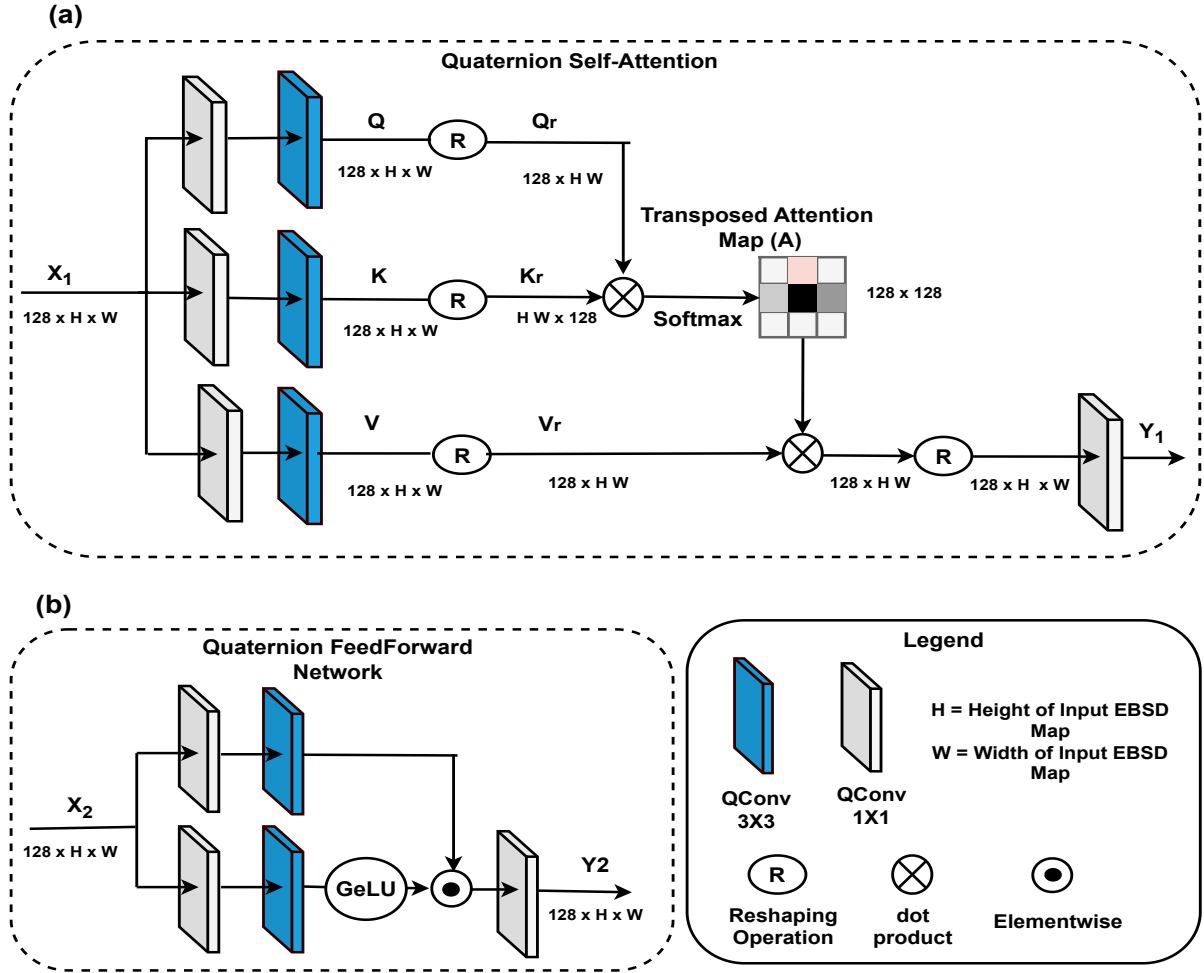


Figure 4.4: QUATERNION SELF ATTENTION: Self-Attention in (a) is computed using quaternion convolution across feature dimension instead of spatial dimension to reduce computational complexity to linear. A transposed attention map (A) is calculated from reshaped query (Q_r) and reshaped key (K_r). A quaternion self-attention is computed from the transposed attention map (A) and reshaped value (V_r). QUATERNION FEED FORWARD NETWORK: Shown in (b), performs controlled feature transformation to allow useful information to propagate further using gated quaternion convolution.

4.2.7 Upsampling and Reconstruction

The upsampling and reconstruction module has 1D pixelshuffle layers and quaternion convolution layers of kernel size 3. The original pixelshuffle layer [89] is designed for 2D upsampling, but we have modified it for 1D upsampling in our framework that generates

information in z dimension. Each block of the upsampling and reconstruction module upsamples deep features by a factor of 2, with the number of blocks depending on the scaling factors.

$$F_{\uparrow} = H_{\uparrow}(F_{DF}) \tag{4.9}$$

$$I_{SR} = H_R(F_{\uparrow}) \tag{4.10}$$

Where, each upsampling block (H_{\uparrow}) of the module has a quaternion convolution layer of kernel size (3×3) as explained in equation 4.2, and a 1D pixel-shuffle layer. The reconstruction block (H_R) is a quaternion convolution layer of kernel size (3×3) .

4.2.8 Activation Function

Introducing non-linearity through an activation function is not straightforward for quaternions, as discussed in [84]. However, quaternion activation functions have been adapted for use in QNNs with standard backpropagation algorithms [90,91]. There are two classes of these quaternion-valued activation functions: fully quaternion-valued functions and split functions. Fully quaternion-valued activation functions are an extension to the hypercomplex domain of real-valued functions, such as sigmoid or hyperbolic tangent functions. Despite their better performance [92], careful training is needed due to the occurrence of singularities that can affect performance. To avoid this, split activation functions [92,93] have been presented as a simpler solution for QNNs. In split activation functions, a conventional real-valued function is applied component-wise to a quaternion, alleviating singularities while holding true the universal approximation theorem as demonstrated in [93]. We have used the split ReLU function which is defined as follows:

$$ReLU(P'_{quaternion}) = \begin{bmatrix} ReLU(P'_R) \\ ReLU(P'_X) \\ ReLU(P'_Y) \\ ReLU(P'_Z) \end{bmatrix} \quad (4.11)$$

4.2.9 2D to 3D EBSD

The output of the QRBSA network is a 2D high-resolution EBSD map from a sparsely sectioned 2D EBSD map in the z direction. The 2D high-resolution EBSD maps are then combined to make a 3D volume. The missing z rows, as in Figure 4.1 (c), can be generated either from xz plane (y_{normal}) or yz plane (x_{normal}). Therefore, there are two ways to form the 3D volume. In this work, we generated both 3D volumes separately, but we plan to design an algorithm in the future to combine the xz plane and yz plane information to make a single 3D volume.

4.2.10 Training Loss for the Network

The QRBSA network is trained using a physics based loss function [72], which uses rotational distance approximation loss with enforced hexagonal crystal symmetry (HCP). Rotational distance loss computes the misorientation angles between the predicted and ground truth EBSD map in the same manner that they would be measured during crystallographic analysis, with approximations to avoid discontinuities at the edge of the fundamental zone. The rotational distance θ between two unit quaternions can be computed as the following:

$$q1 \cdot q2 = \cos(\theta) \longrightarrow \theta = 4 \sin^{-1} \left(\frac{d_{euclid}}{2} \right) \quad (4.12)$$

where, $d_{\text{euclid}} = \|q_1 - q_2\|_2$. While d_{euclid} is Lipschitz, the gradient of θ goes to ∞ as $d_{\text{euclid}} \rightarrow 2$. To address this issue during neural network training, a linear approximation was computed at $d_{\text{euclid}} = 1.9$, and utilized for points > 1.9 .

4.2.11 Network Implementation and Output Evaluation

We use a learning rate of 0.0002, an Adam optimizer with $\beta_1 = 0.9$, $\beta_2=0.99$, ReLU activation, batch size of 4 and downscaling factor of 2 and 4. The patch size of HR EBSD maps is selected from $\{16, 32, 64, 100\}$ during training of the network. The framework is implemented in PyTorch and trained on NVIDIA Tesla V100 GPU for 2000 epochs, which took approximately 100 hours. Once training is completed, inference time for a given 2D LR EBSD map is on the order of less than one second for an imaging area that would normally take about 10 minutes to gather manually.

4.2.12 Progressive Learning

In chapter 3, we used a fixed patch size of dimension 64×64 for training the CNN based architectures which help in learning local correlations. However, self-attention is required to have larger patch sizes, which aids in learning global correlations. Inspired from the work of Zamir [71], we use progressive patch samples from sizes of $\{16, 32, 64, 100\}$ in the training process to learn global statistics. We start from a smaller patch size in early epochs and increase to a larger patch sizes in the later epochs. The progressive learning acts like the curriculum learning process where a network starts with a simple tasks and gradually moves to learning a more complex one.

4.3 Results

4.3.1 Qualitative Output Comparison

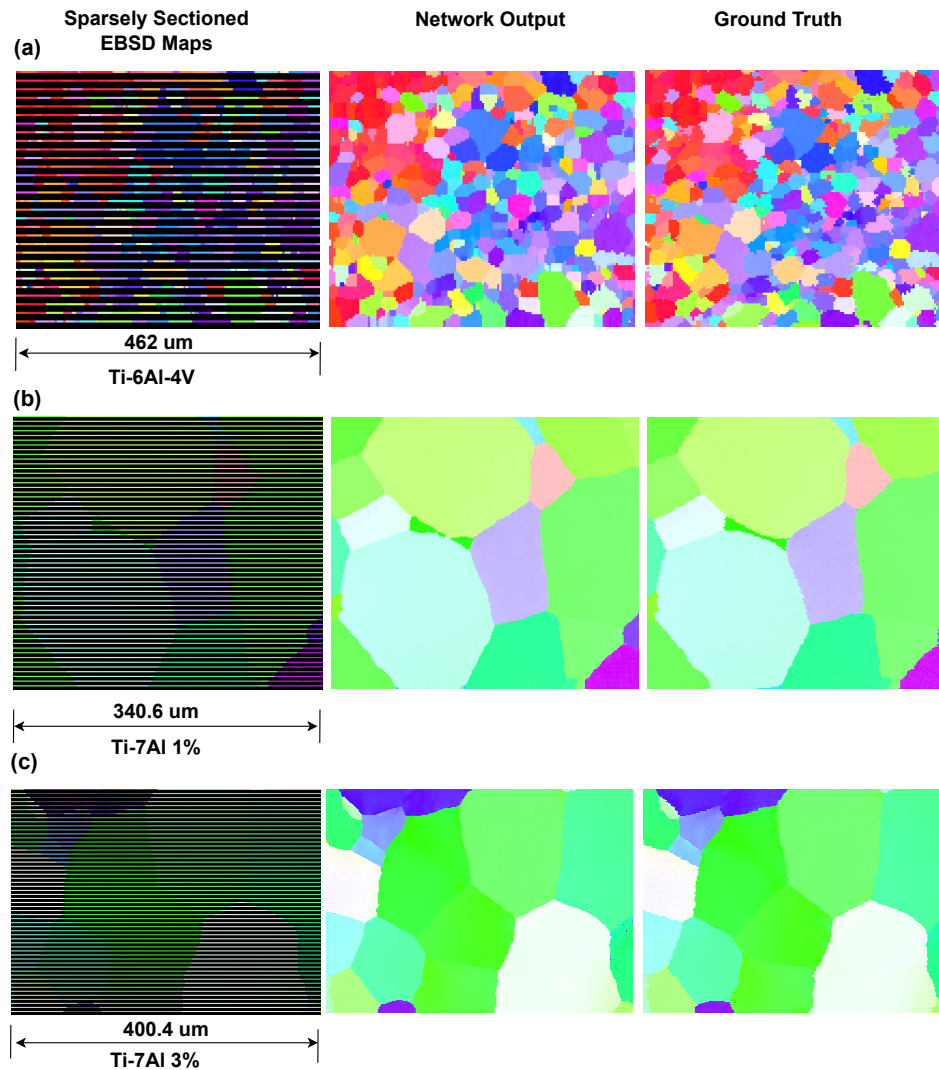


Figure 4.5: VISUAL COMPARISON OF NETWORK OUTPUT FOR EXAMPLE 2D EBSD MAPS WITH A SCALE FACTOR OF 4: The predicted EBSD maps (Network Output) from the QRBSA network are similar to the ground truth EBSD maps in for both the Ti-6Al-4V dataset (a) and both Ti-7Al datasets (b) and (c). The black rows correspond to the missing data in the sparsely sectioned input EBSD maps. In this case, one row of EBSD data is used for every three rows of missing EBSD data.

The sparsely sectioned 3D EBSD data is downsampled by scale factors of 2, 4 in the

z dimension by removing the xy planes (z_{normal}) to reflect how EBSD resolution would be reduced in a serial sectioning experiment. Our network QRBSA is trained on 2D orthogonal planes (x_{normal} and y_{normal}) of paired sparsely sectioned EBSD maps and high-resolution EBSD maps, generating the high-resolution 2D maps in z dimension shown in Figure 4.5. The most noticeable visual defects in 2D appear as pixel noise or short vertical lines, particularly around small grain features and high-aspect-ratio grains whose shortest axis is aligned with the z -direction. In addition to planar output analysis, we can also create 3D volumes from the sparsely sectioned xz planar (y_{normal}) or yz planar (x_{normal}) EBSD maps, and then sample the xy planes (z_{normal}) from these volumes, as represented by the black arrows in Figure 4.6, to evaluate how well the QRBSA is inferring missing z -sample planes. Note that the planes visualized in Figure 4.6 are not immediately adjacent to any ground truth planes. We can observe that our deep learning framework is able to completely predict omitted xy planes, comparably to the ground truth xy plane, with the exception of some shape variations around grain boundaries, particularly in Ti-6Al-4V. We capture these errors in Figure 4.6 in the column labeled misorientation angle map, which is contrast scaled such that all misorientation errors greater than 3° appear as white. Looking at this map, most of the high misorientation errors are at grain boundaries with the exception of some specific small grains in Ti-6Al-4V. Observations of this difference map indicate that if the xy plane in Figure 4.6 had been omitted during experimental data collection, our framework would have estimated it with reasonable accuracy.

4.3.2 Quantitative Output Comparison

The pixel-wise distribution of minimum misorientations between network output and ground truth, referred to as misorientation error, is shown in Figure 4.7. The x-axes of

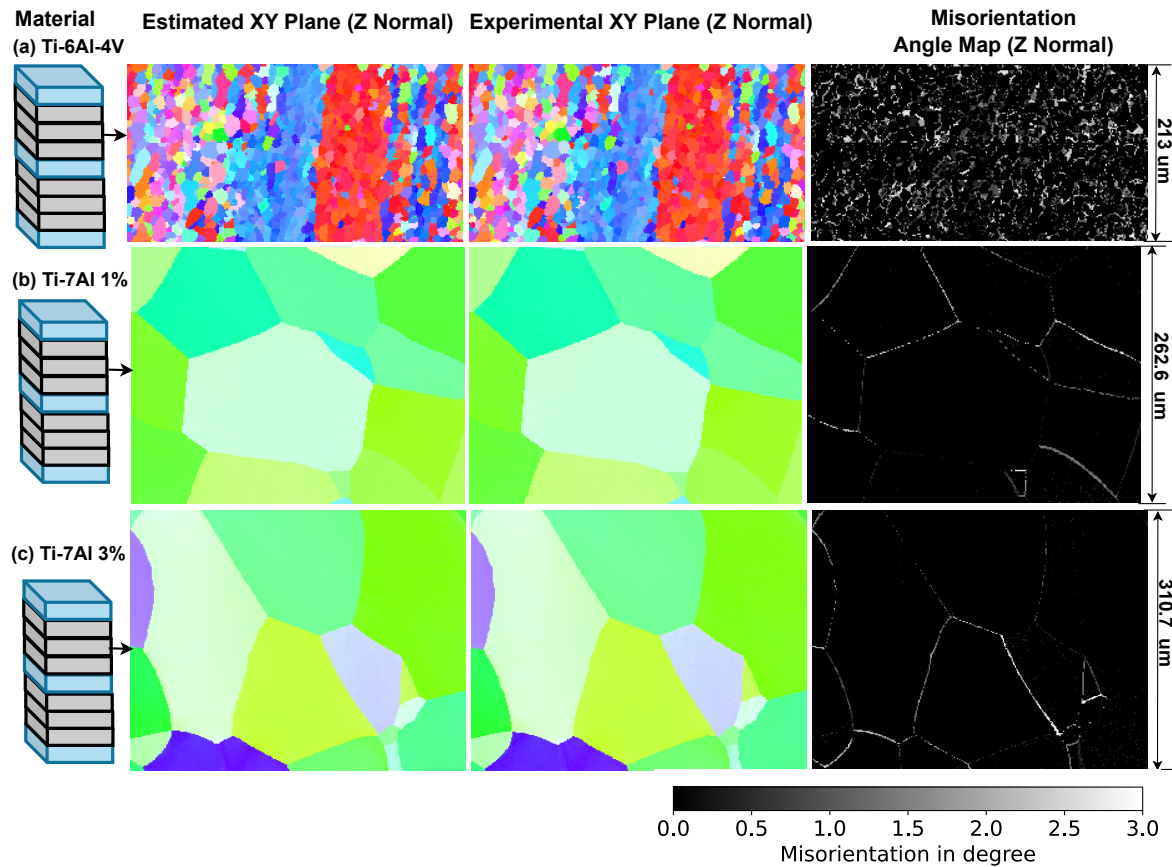


Figure 4.6: NEURAL NETWORK OUTPUT VS. GROUND TRUTH WITH DIFFERENCE MAP: The deep learning framework is able to estimate the missing xy planes due to sparse z-sampling (gray) with data that looks similar to the ground truth for Ti-6Al-4V in (a) and Ti-7Al in (b) and (c). The misorientation angle map column shows the minimum possible misorientation between ground truth and estimated EBSD maps with 3° thresholded maximum to better show low magnitude errors. This map indicates that learning grain shapes for Ti-6Al-4V is more difficult than for Ti-7Al, likely due to smaller grain size and more grain boundary regions.

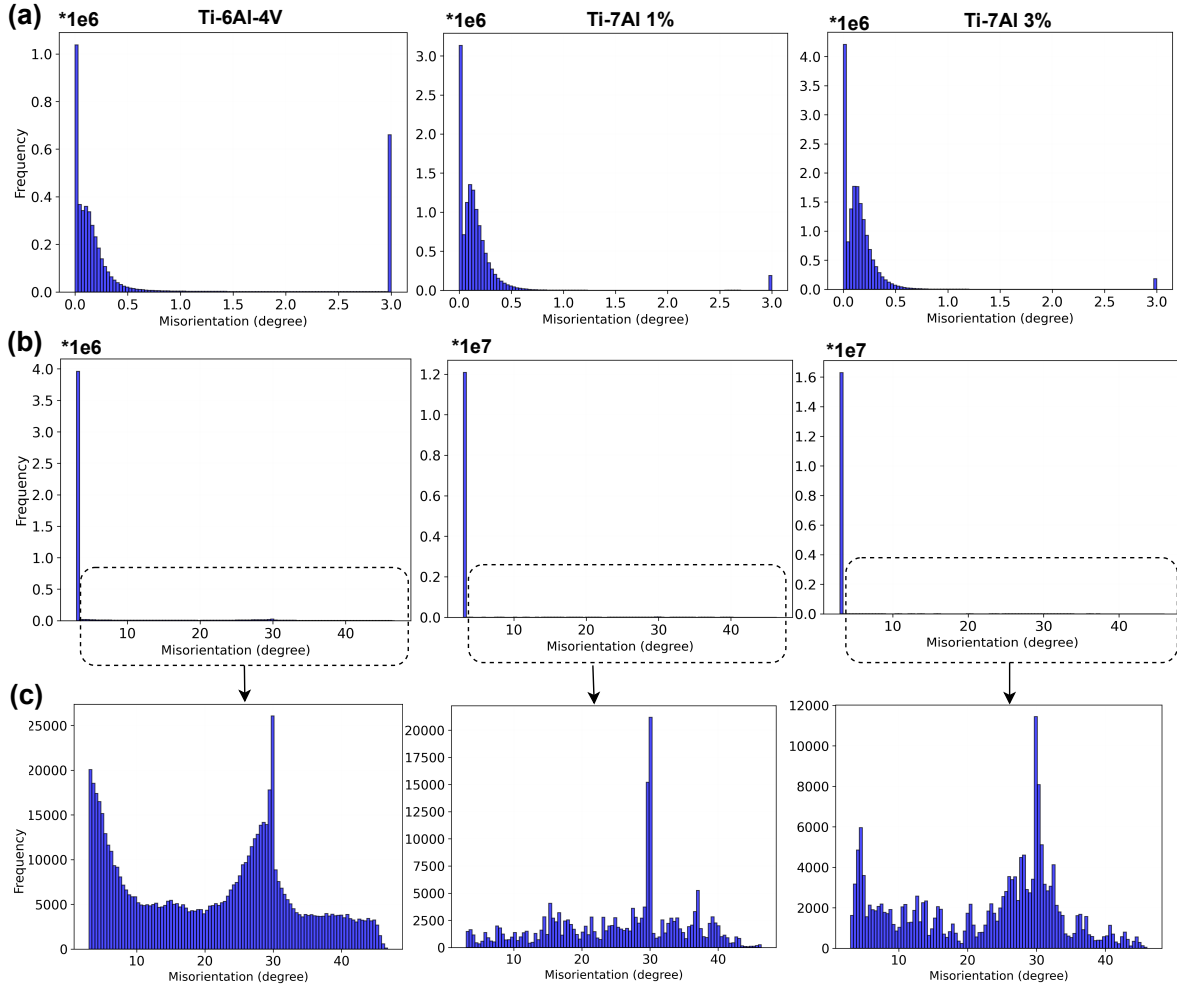


Figure 4.7: HISTOGRAM OF MISORIENTATION ANGLE FOR Ti-6Al-4V, Ti-7Al 1%, AND Ti-7Al 3%: In (a), histograms of misorientation differences between predicted and ground truth are shown, where all values greater than 3° are clamped to 3° . For all materials, most network error in predicted misorientation is lower than 0.5° in magnitude. In (b), the same error histograms are displayed, but now misorientation values less than 3° are clamped to 3° . Because larger magnitude errors occur far less frequently than smaller errors, (c) contains a zoomed inset of misorientation angles greater than 3° to better show their distribution.

the histograms are thresholded and separated at 3° , such that misorientation error of magnitude less than 3° is shown in Figure 4.7(a), and error greater than 3° is shown in Figure 4.7(b) and (c). These histograms show that the majority of network error

is relatively unimodal and smaller in magnitude than about 0.4° , meaning that it will fall primarily within the dark regions of the grains in the difference maps in Figure 4.6, which correspond to small intragranular misorientation errors. On the other hand, most of the high misorientation errors in Figure 4.6 are much larger than 3° , which mostly correspond to errors in predicted grain boundary location, or small grains that were ill-defined in the low resolution input. While these errors are much larger in magnitude, Figure 4.7(b) shows that these represent a very small fraction of network error. A more detailed inspection of this error in Figure 4.7(c) shows that this larger error is relatively random and uniform, with the exception of a spike around 30° , which can be seen in all three datasets. This spike in error around 30° may be related to the hexagonal symmetry of the titanium materials, as 30° is a high symmetry rotation within the $6/mmm$ point group, but even so, these errors represent less than 2% of the total.

Network	Trainable Parameters	Ti-6Al-4V		Ti-7Al 1%		Ti-7Al 3%	
		x2	x4	x2	x4	x2	x4
HAN	63,315,578	26.12	25.64	33.67	33.55	35.10	34.36
EDSR	6,355,460	26.70	26.25	34.39	34.25	36.19	36.05
QEDSR	1,593,092	26.62	26.16	34.23	34.10	35.56	35.44
QRBSA	5,952,782	27.71	27.29	35.29	35.13	36.64	36.52

Table 4.1: **PSNR:** COMPARISON OF PSNR OF MISORIENTATION ANGLE AND COMPLEXITY FOR DIFFERENT NETWORKS FOR SCALE FACTORS 2 AND 4: Columns represent number of trainable parameters and PSNR for different titanium datasets. A larger number is desired for both PSNR.

The peak signal to noise ratio (PSNR) of misorientation angle between ground truth and experimental EBSD data is shown in Table 4.1 to quantitatively evaluate the performance of the QRBSA network for scale factors 2 and 4 for all three materials. Higher PSNR values represent more similarity with the ground truth. The PSNR of Ti-6Al-4V is lower compared to Ti-7Al datasets due to its higher texture variability, wider range of orientations, and generally smaller grain features. We performed this same analysis on

four different network architectures with different computational complexity, as shown in Table 4.1.

When considering the relationship between network complexity and performance for this use-case, a simpler deep residual architecture (EDSR) [38], outperforms a more complex holistic attention network (HAN) [41] on EBSD data despite having significantly lower computational complexity. The amount of available EBSD data in this case is significantly lower than open-source RGB image datasets, so simply increasing network complexity does not improve performance, as this added complexity demands additional training information and does not meaningfully consider relevant data modalities. QEDSR incorporates quaternion considerations in a similar architecture to EDSR, which greatly reduces in the number of network parameters, but also causes a slight drop in performance due to overall lack of complexity. We take advantage of this reduction in complexity to add in additional self-attention for better recognition of long-range patterns and global statistics. This QRBSA network demonstrates the best performance on EBSD map restoration, while still maintaining lower complexity than state-of-the-art residual architectures for single-image super-resolution tasks.

4.3.3 Ablation Study

To show the effectiveness of quaternion convolution, we have done experiments with an enhanced deep scale residual (EDSR) network [38] and a quaternion enhanced deep residual (QEDSR) network as shown in Figure 4.10. The QEDSR is similar to EDSR except that the real-valued convolution layer is replaced by quaternion convolution layer that helps in reducing computational complexity. With decreased complexity due to the quaternion convolution layer, we can add a quaternion transformer block into our network architecture to learn global features of EBSD maps. The PSNR/SSIM [57]

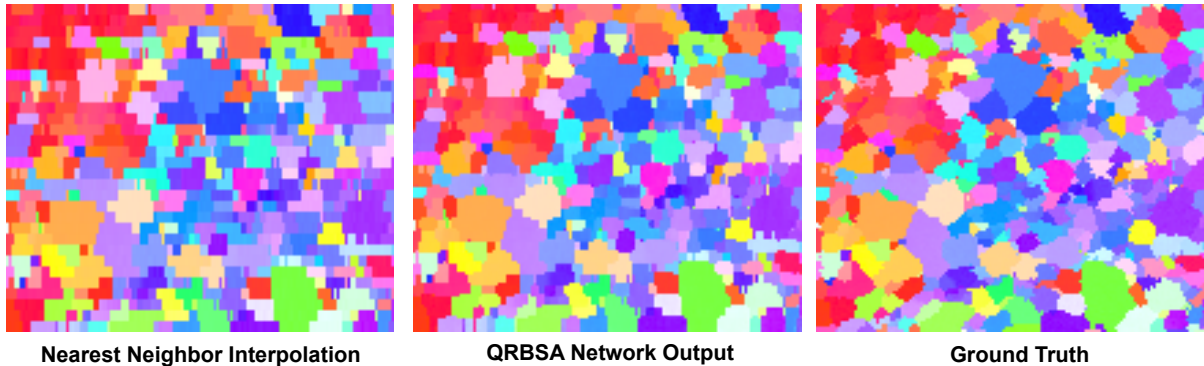


Figure 4.8: COMPARISON OF NETWORK OUTPUT WITH NEAREST NEIGHBOUR: Nearest neighbor interpolation method gives blocky grains and visually identical results to sparsely sectioned EBSD data.

values of IPF map for the generated EBSD maps and number of trainable parameters of different neural network architectures are shown in Table 4.2. Our network QRBSA is giving better PSNR and SSIM than EDSR [38], while having fewer trainable parameters. To understand the effect of the quaternion transformer block in QRBSA, we have trained our network with and without the quaternion transformer block as shown in Table 4.3. We see an improvement in PSNR and SSIM with the quaternion transformer block for the Ti6-Al-4V dataset, which has more crystallographic texturing and is difficult to learn because of the mismatched lengthscales over which it persists, compared to the grain structure. The quaternion transformer block will become even more useful for learning global features as we increase our library of 3D datasets available to our deep learning framework in future.

During analysis of our results, we found that some of generated xy planes (z_{normal}) in all three of the investigated datasets (Ti-6Al-4V, Ti-7Al 1 %, and Ti-7Al 3 %) have minor noise, as shown in Figure 4.9. In future work, we will be focusing on reducing this noise in the estimated xy planes by using 3D neural network architecture.

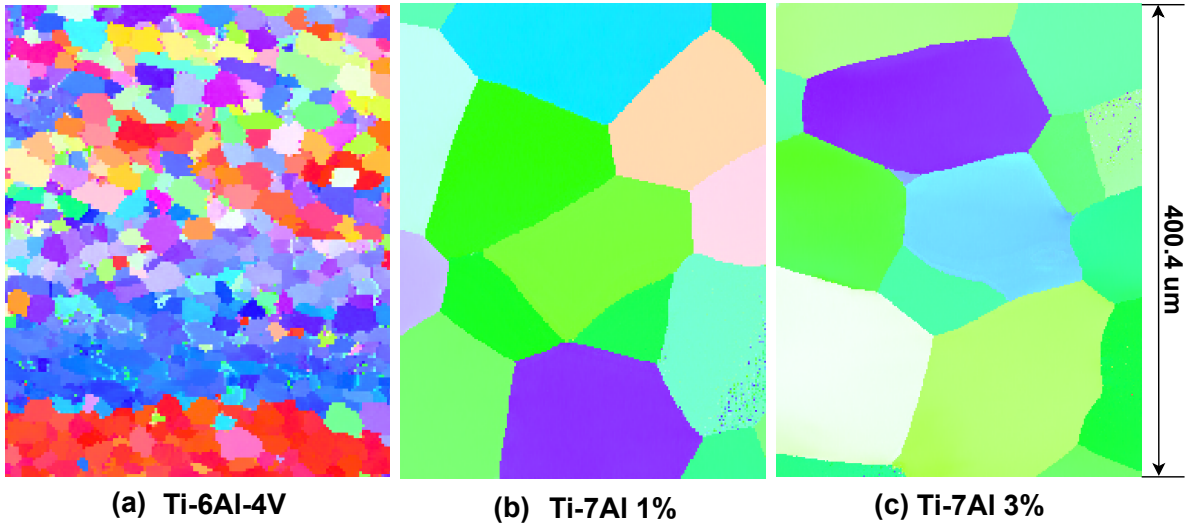


Figure 4.9: NOISE IN ESTIMATED XY PLANE: The deep learning framework is able to estimate the missing xy planes in z dimension but there are some minor noises in some of the xy planes.

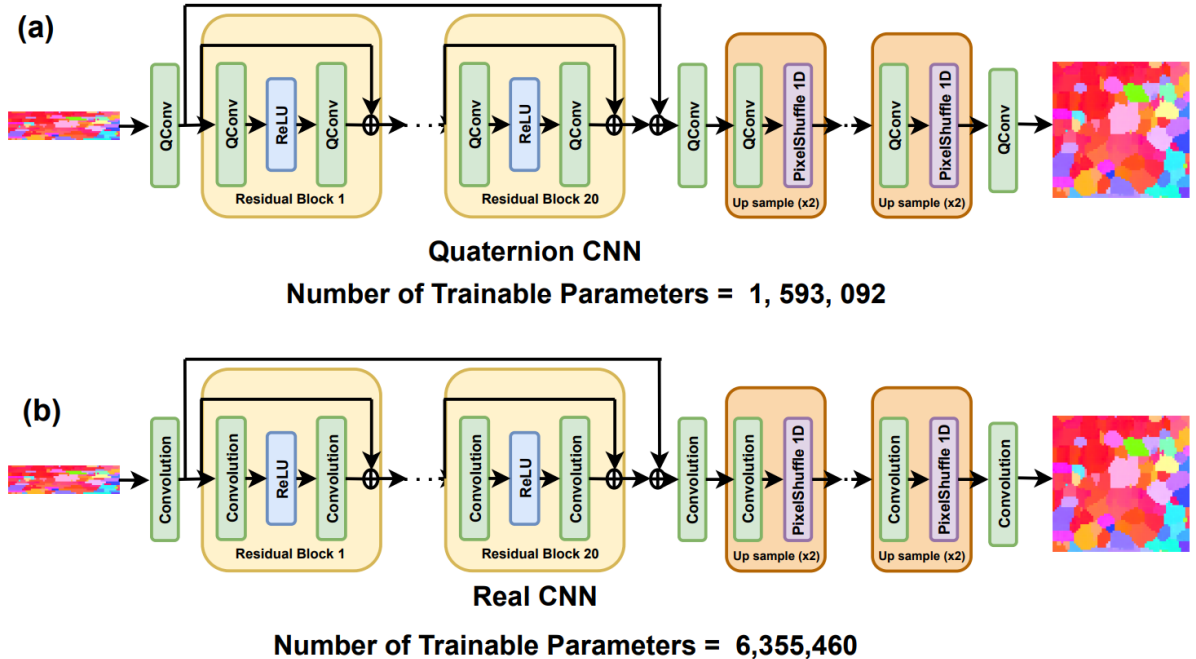


Figure 4.10: QUATERNION CNN AND REAL CNN: Both QCNN and RCNN have same number of layers except basic convolution operation layer. In QCNN, total number of trainable parameters are reduced significantly which give us room to add more complexity such self-attention layer to learn global features.

Network	Trainable Parameters	Ti-6Al-4V		Ti-7Al 1%		Ti-7Al 3%	
		x2	x4	x2	x4	x2	x4
HAN	63,315,578	21.21/ 0.839	17.55/ 0.673	29.36/ 0.919	26.54/ 0.849	31.26/ 0.941	28.90/ 0.905
EDSR	6,355,460	21.49/ 0.86	18.09/ 0.718	30.06/ 0.944	27.34/ 0.907	32.16/ 0.958	29.57/ 0.932
QEEDSR	1,593,092	21.44/ 0.861	18.04/ 0.710	29.89/ 0.935	27.22/ 0.904	32.05/ 0.95	29.52/ 0.93
QRBSA	5,952,782	21.60/ 0.870	18.2/ 0.730	30.20/ 0.946	27.48/ 0.908	32.36/ 0.96	29.65/ 0.94

Table 4.2: **PSNR/SSIM of IPF Maps:** COMPARISON OF PSNR/SSIM OF IPF MAPS AND COMPLEXITY FOR DIFFERENT NETWORKS FOR SCALE FACTORS 2 AND 4: Columns represent number of trainable parameters and PSNR/SSIM for different titanium datasets. A larger number is desired for both PSNR/SSIM.

Experiments	Ti-6Al-4V	Ti-7Al 1%	Ti-7Al 3%
QRBSA without quat transformer	18.12/ 0.718	27.27/ 0.906	29.49/ 0.9314
QRBSA with quat transformer	18.2/ 0.73	27.48/ 0.908	29.65/ 0.94

Table 4.3: **PSNR/SSIM of IPF Maps:** COMPARISON OF PSNR/SSIM OF IPF MAPS WITH AND WITHOUT QUATERNION TRANSFORMER FOR SCALE FACTOR 4: Columns represent PSNR/SSIM for different titanium datasets. A larger number is desired for both PSNR/SSIM.

4.4 Q-RBSA BisQue Module

We have successfully integrated the QRBSA inference module into the BisQue platform, as depicted in the Figure 4.11. To utilize this feature, users are required to upload sparsely sectioned EBSD data in numpy format and then click the 'Run' button. The module efficiently processes this input to produce three key outputs: Superresolved EBSD data, also provided in numpy format, and HDF5 files corresponding to both the sparsely sectioned input and the generated superresolved file. These HDF5 files can be readily visualized on the BisQue platform, offering users a seamless and intuitive interface for both input and output data, as demonstrated in the accompanying Figures 4.12 and 4.13.

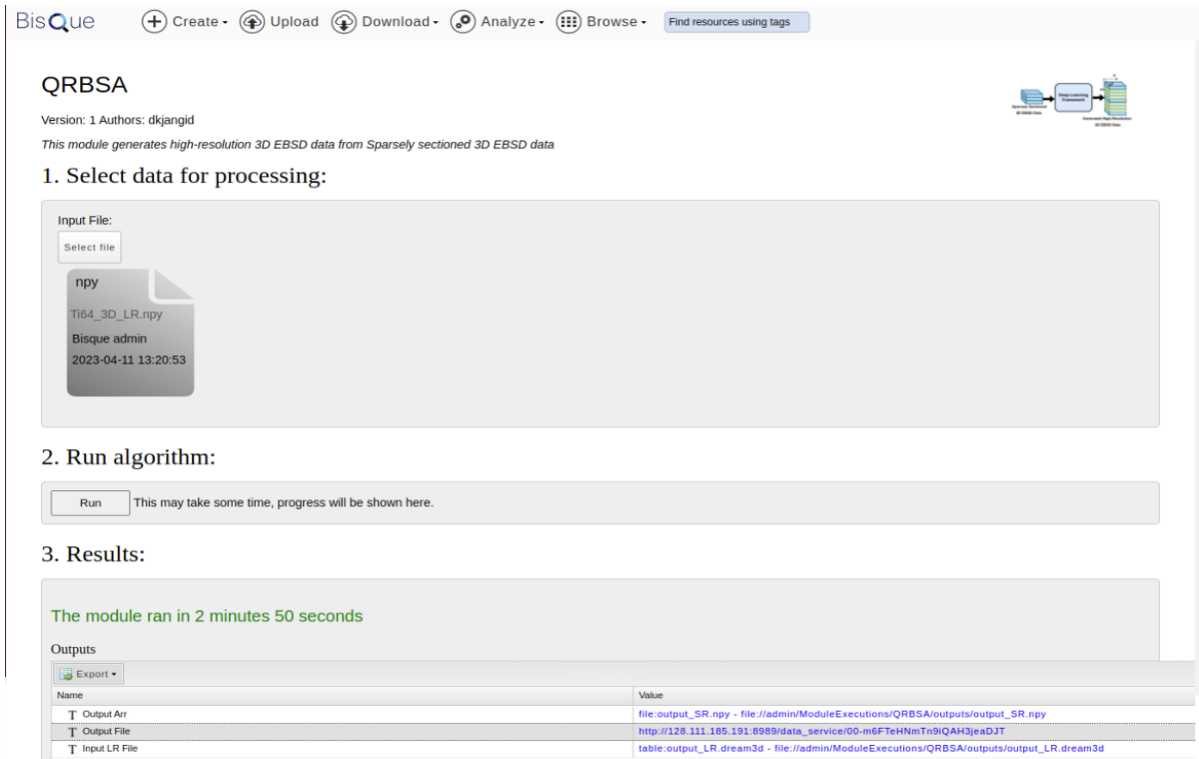


Figure 4.11: BISQUE UI INTERFACE A user interface of generating high-resolution 3D Microstructure from given sparsely sectioned 3D Microstructure

Architecture code is publicly accessible through GitHub (<https://github.com/UCSB-VRL/Q-RBSA>).

4.5 Discussion and Conclusion

Both quantitative and qualitative results demonstrate that this physics-based deep learning framework can accurately estimate the missing xy planes (z_{normal}) of 3D EBSD data for multiple variants of titanium alloys, both with a coarser polycrystalline structure (Ti-7Al) and finer structure with stronger texture (Ti-6Al-4V). In 2D inferred EBSD planes show noise around small features, mostly in the form of point and line defects in the z -direction associated with grains whose overall shape information was lost due to

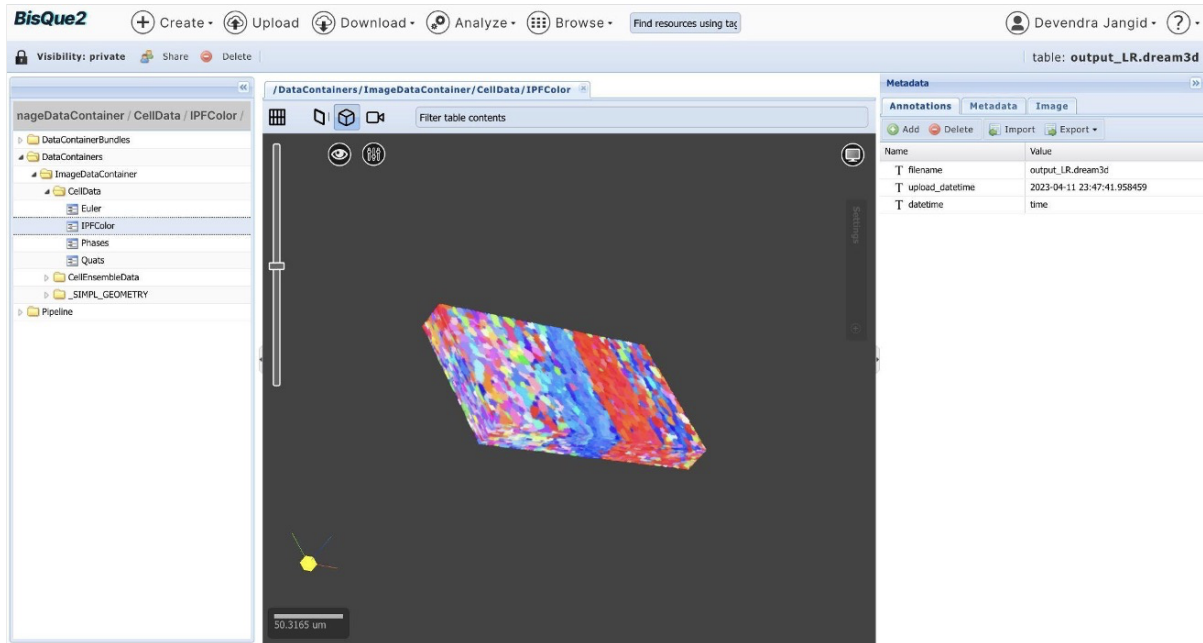


Figure 4.12: SPARSELY SECTIONED MICROSTRUCTURE: Visualization of a sparsely sectioned microstructure on BisQue

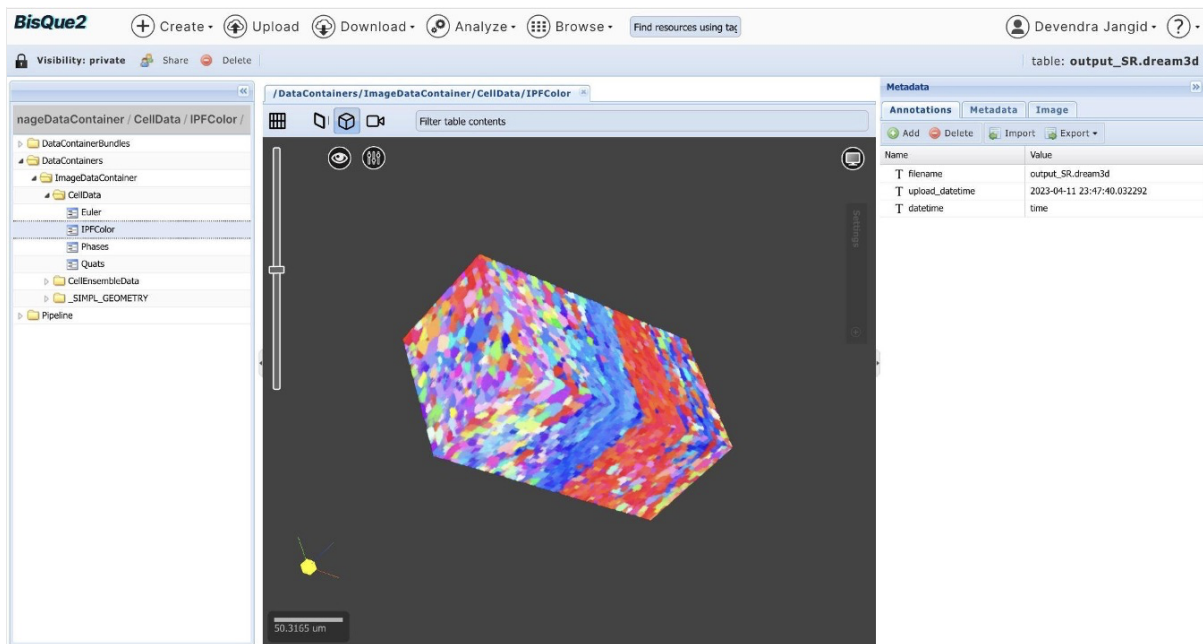


Figure 4.13: HIGH RESOLUTION MICROSTRUCTURE: Visualization of generated high resolution microstructure

omission of sample planes in low resolution. It is possible that a downsampling approach incorporating anti-aliasing could prevent this shape information loss [57], but this approach would not be reflective of actual experimental downsampling in 3D EBSD. This general shape loss effect, along with a larger number of small grains, varying local crystallographic texture, and a wide range of represented crystal orientations, made the Ti-6Al-4V the most difficult dataset for inference. This is further evidenced by a larger number of grain boundary differences for Ti-6Al-4V in Figure 4.6, as well as a lower PSNR score in Table 4.1. Additional noise analysis for generated xy planes is shown in the Figure 4.9. In the future, 3D architectures and grain shape information [94] with adaptive multi-scale imaging in z dimension can be used to improve performance as more of this type of data becomes available. In this chapter, fixed scaling factors were employed to downsample microstructure data. Moving forward, it will be essential to implement adaptive scaling factors. This approach will enable the collection of high-resolution data at smaller step sizes in areas where critical characteristics significantly influence the material's properties. Conversely, larger step sizes can be utilized in regions where the information repeats and does not require high granularity. This method ensures a more efficient and precise analysis of the microstructure, focusing on areas of importance while conserving resources in less critical sections.

The limiting factor when using the network approach presented here on serial-sectioned 3D microstructures is the ratio of serial sectioning spacing in the low-resolution input relative to the size of the microstructural features being imaged. For example, if the serial section spacing is large enough to skip entire grains or microstructural features in a material, those features will never be resolvable with super-resolution. Therefore, an informed super-resolution scaling factor choice must be made prior to any experiment to ensure that the low resolution input contains enough information for meaningful inference. Beyond this section depth limitation, the approach shown here is directly applicable

to any serial sectioning technique for gathering 3D EBSD information, including FIB sectioning, laser ablation, and robotic serial sectioning [65,66]. Further, data from other 3D grain mapping techniques that rely on synchrotron X-ray sources such as diffraction contrast tomography (DCT) [64] or high energy diffraction microscopy (HEDM) [62,63] may also be applicable for the infrastructure presented here. Similar approaches to this may be particularly useful in lab source DCT experiments [95,96], where the X-ray source constraints limit grain mapping resolution in comparison to synchrotron sources. For example, one could use difficult to acquire synchrotron X-ray mapping experiments as HR data to train a network to inform LR X-ray mapping experiments collected more routinely at the laboratory.

In summary, we have designed a quaternion-convolution-based deep learning framework with crystallography physics-based loss to generate costly high-resolution 3D EBSD data from sparsely sectioned 3D EBSD data while accounting for the physical constraints of crystal orientation and symmetry. Alongside this, an efficient quaternion-based transformer block was developed to learn long-range trends and global statistics from EBSD maps. Using quaternion convolution instead of regular convolution is critical for crystallographic data, both in terms of output quality and neural network complexity, as reducing the number of trainable parameters enables transformer addition without major complexity burden (see Table 4.1). This framework can be directly applied to any experimental 3D EBSD approaches that rely on serial sectioning techniques to collect orientation information. In this thesis, fixed scaling factors were employed to downsample microstructure data. Moving forward, it will be essential to implement adaptive scaling factors. This approach will enable the collection of high-resolution data at smaller step sizes in areas where critical characteristics significantly influence the material's properties. Conversely, larger step sizes can be utilized in regions where the information repeats and does not require high granularity. This method ensures a more efficient and precise

microstructure analysis, focusing on important areas while conserving resources in less critical sections.

4.6 Appendix

4.6.1 Quaternion Background

Quaternion Number System: A quaternion q is a four component number of the form $q = q_0 + iq_1 + jq_2 + kq_3$, where basis vectors (i, j, k) satisfy the following relationship:

$$i^2 = j^2 = k^2 = ijk = -1 \tag{4.13}$$

The quaternion can be represented as matrix of real numbers.

$$\begin{bmatrix} q_0 & q_1 & q_2 & q_3 \\ -q_1 & q_0 & -q_3 & q_2 \\ -q_2 & q_3 & q_0 & -q_1 \\ -q_3 & -q_2 & q_1 & q_0 \end{bmatrix} \tag{4.14}$$

Quaternion Convolution: The difference between real-valued convolution neural networks (RCNNs) and quaternion convolution neural networks (QCNNs) lies in their approach to performing basic convolution operations, as illustrated in Figure 4.14. In RCNNs, scalar feature maps are computed by independently correlating channels of feature maps with kernel filters, without considering inter-channel relationships. However, in QCNNs, vector feature maps are generated using the Hamilton product between feature maps and quaternion kernels. Both quaternion kernels and Hamilton products take inter-channel relationships into account in quaternion convolutional neural networks. As a re-

sult, for the same number of output channels, quaternion convolution requires one-fourth of the kernel filters used in real-valued convolution, resulting in a four-fold reduction in parameters. Quaternion output feature maps undergo a pooling and concatenation operation to rearrange scalar and vector components of quaternion feature maps, as shown in Figure 4.15. The concatenated quaternion feature maps are then fed into the second layer to compute the Hamilton product with quaternion kernel filters, as shown in Figure 4.16. This process is repeated for further layers.

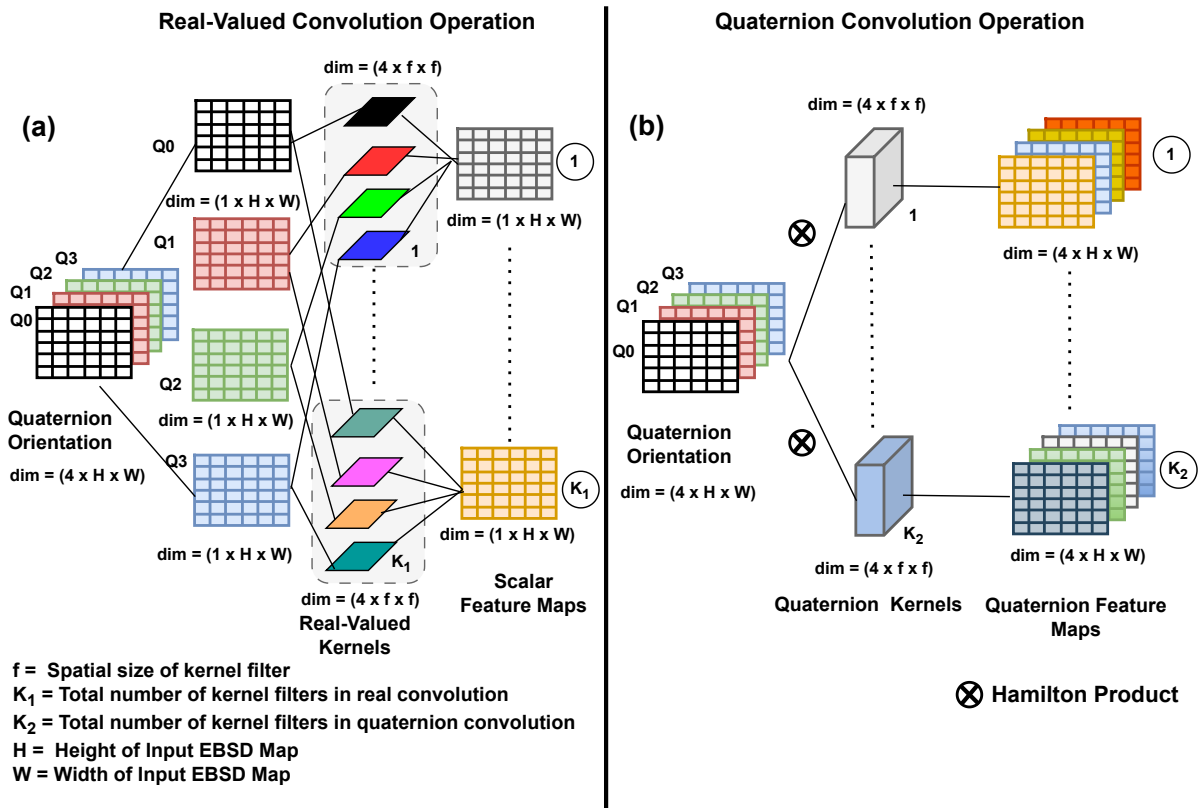


Figure 4.14: DIFFERENCES BETWEEN REAL-VALUED CONVOLUTION AND QUATERNION CONVOLUTION: Real-valued convolution operations compute correlations, which are scalar quantities, between independent kernels and quaternion input feature maps. In contrast, quaternion convolution operations compute the Hamilton product, which is a vector quantity, between quaternion kernels and quaternion input feature maps. Furthermore, quaternion convolution requires only one-fourth of the kernel filters needed in real-valued convolution for the same number of output channels, resulting in a substantial reduction in total kernel filters.

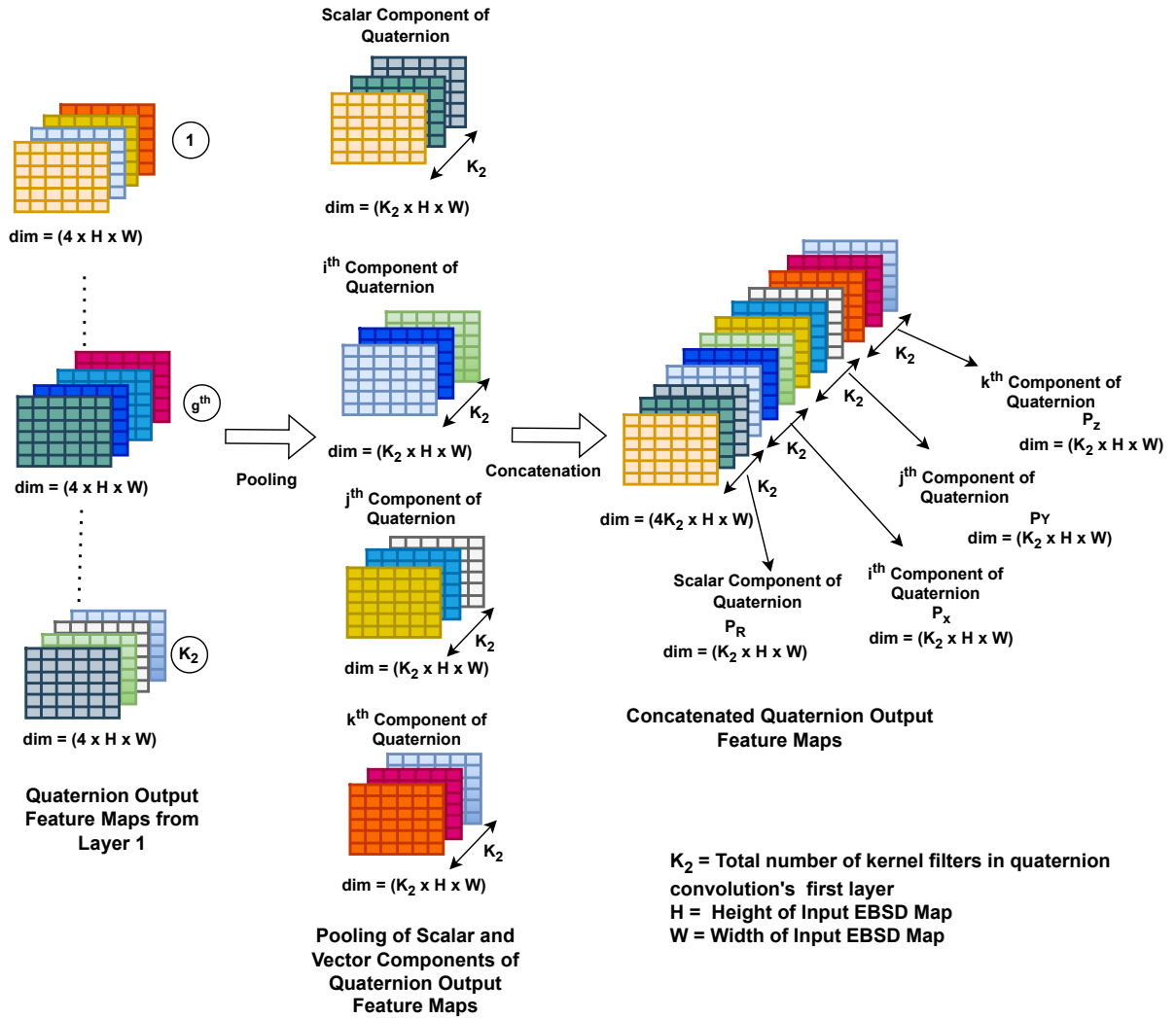


Figure 4.15: POOLING AND CONCATENATION OF QUATERNION FEATURE MAPS: Following the first quaternion convolution layer with K_2 filters, as illustrated in Figure 4.14, the resulting feature maps are pooled into scalar and vector components (i^{th} , j^{th} , k^{th}) of Quaternion Feature Maps, with each component having a K_2 channel dimension. These pooled features are then concatenated along the channel dimension, resulting in a total of $4K_2$ channel dimensions.

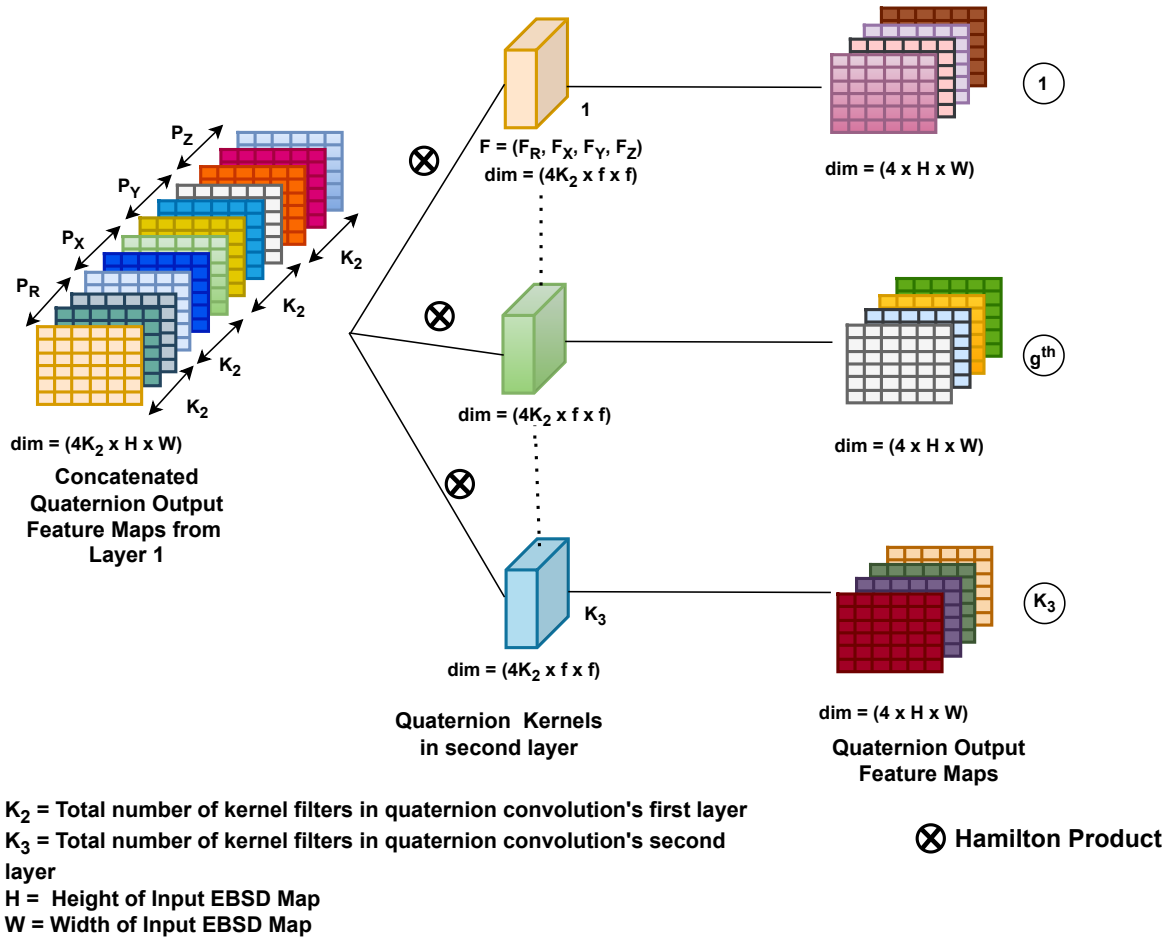


Figure 4.16: EXPLANATION OF SECOND LAYER IN QUATERNION CONVOLUTION NEURAL NETWORK: In the second layer of the quaternion neural network, the Hamilton product is computed between the output of the pooling and concatenation operation, as shown in Figure 4.15, and K_3 quaternion kernel filters. This results in K_3 quaternion output feature maps, each with a dimension of $(4, H, W)$.

Chapter 5

3D Grain Shape Generation in Polycrystals Using Generative Adversarial Networks

Generative AI models significantly enhance materials discovery by overcoming the challenges of slow and expensive experimental processes. By generating detailed 3D synthetic microstructures, these models rapidly explore complex process-structure-property relationships. Utilizing advanced algorithms and computational tools, they create accurate representations of materials, providing a quicker, more cost-effective way to predict material behaviors and properties.

In this chapter, we present a Generative Adversarial Network (GAN) capable of producing realistic microstructure morphology features, a primary step to construct 3D synthetic microstructure, and demonstrates its capabilities on a dataset of crystalline titanium grain shapes. Alongside this, we present an approach to train deep learning networks to understand material specific descriptor features, such as grain shapes, based on existing conceptual relationships with established learning spaces, such as functional

object shapes. A style-based GAN with Wasserstein loss, called M-GAN, is first trained to recognize distributions of morphology features from function objects in the ShapeNet dataset, and is then applied to grain morphologies from a 3D crystallographic dataset of Ti-6Al-4V. Evaluation of feature recognition on objects showed comparable or better performance than state-of-the-art voxel-based network approaches. When applied to experimental data, M-GAN generated realistic grain morphologies comparable to those seen in Ti-6Al-4V. A quantitative comparison of moment invariant distributions showed that the generated grains were similar in shape and structure to the ground truth, but scale invariance learned from object recognition led to difficulty in distinguishing between the physical features of small grains and spatial resolution artifacts. The physical implications of M-GAN's learning capabilities are discussed, as well as the extensibility of this approach to other material characteristics related to grain morphology. The contents of this chapter are discussed in our published paper [94].

5.1 Introduction

One of the most pervasive challenges of materials discovery and development is that the process demands an enormous amount of time, labor, and capital. Reducing these demands has been a central driver for a number of international initiatives ([97–103]). This challenge stems from the inefficient use of a linear approach to explore high-dimensional materials spaces, which is compounded by critical metrics for material classes that account for both processing variations as well as complexity of application environment. For example, there are a wide range of metal alloy compositions that can be used in turbine engine components, and the type of alloy being used can vary widely depending on the location and operating demands of the specific turbine section. But regardless of composition chosen, any alloy must be able withstand the operating environment, and

evaluations of its performance must cover thousands or even millions of manufacturing iterations. The sheer size of these composition and application spaces places materials science at a unique tipping point for the integration of machine learning to rapidly accelerate design and discovery.

Deep learning methods have attracted attention as cost reducers in fields such as voice, text, and image processing. Neural networks perform strongly both in accuracy and efficiency when trained on large collections of well-labelled ground-truth data. These networks may reliably outperform traditional algorithmic approaches and can make further important distinctions, like the identification of specific voices, faces, or fingerprints, with only a small amount of additional input. Neural networks can accelerate and refine the research process by reducing the quantity of required experimental iterations and by enhancing the accuracy and efficiency of material analyses. However, their performance depends on the quality of material data available for training. For materials data in particular, providing sufficient amounts of meaningful information is difficult, and finding ways to address this challenge is the goal of this study.

When considering available material data for use in deep learning approaches, the most significant obstacles are limitations in data scope and dimensionality. Here, data scope is defined as the breadth of readily available data, and what range of information that data describes. Large amounts of materials data have been captured, but much of this data is either difficult to access or is tailored to a specific purpose that limits its broader context. In other deep learning applications, such as text and voice, a great deal of data is made available through literary and audio collections and social media. In materials, not all data is readily available due to the expense of data generation or proprietary concerns, and when data is available, the wide array of collection methods can make establishing broader connections challenging.

With regard to data dimensionality, limitations can be considered in terms of both

variable and spatial dimensions. Variable dimensions refers to the number of different descriptors or characteristics represented in a given dataset. This challenge is particularly prevalent in engineering materials like the ones considered in this investigation. The vast majority of engineering materials are polycrystalline in character, so the bulk material is typically composed of billions of crystals that are rarely uniform in their crystallographic, morphological, or chemical properties. The effects of external environment and test conditions (e.g. specimen geometry, surface treatment, loading conditions) must also be considered, which adds further complexity to these material descriptors. In addition to the number of data variables, it is also critical to consider the dimensions of physical space being represented. In materials analysis, an important point to note is the difficulty of generating and subsequent lack of three-dimensional experimental measurements, which are essential for physically accurate neural network training. Recent experimental advances have greatly improved volumetric materials characterization ([62,66,104–109]), but the quality and quantity of available data are limited by inherent restrictions, such as costly and specialized equipment, reduced spatial/temporal resolution, limitations on sample material/geometry, and sample destruction during characterization.

Deep learning capabilities that have been developed in other domains can be leveraged to address the problem of limited data in the materials domain. Adapting these capabilities for the characteristics of materials requires establishing a relationship between the processing-structure-properties space of materials science and the classification spaces used in other fields of deep learning. One such established space in deep learning is object recognition, where semantically labeled objects are recognized by neural networks based on the collection of features that compose that object. Approaches in this space are readily extensible to materials structure, where relationships can be established between microstructural morphology and other image recognition approaches that focus specifically on object shape. For example, in polycrystalline metals, much of the relationship

between material processing, structure, and properties manifests through the shape and arrangement of the crystalline subdomains we call grains. In the case of uniform crystallization, the physics are well understood. Grains take the shape of faceted polyhedrons, where the number of facets is dictated by impinging neighbors, and facet size and shape are dictated by interfacial energies. Morphologically, these polyhedral shapes are relatively simple, but variations in thermal and mechanical processing can drastically alter the distribution of possible shapes, as demonstrated in Figure 5.1. Under normal growth conditions, the most probable grain shapes are polyhedrons, but orientationally biased mechanical or thermal processing can create conditions where anisotropic grains that are flattened, twisted, or directionally aligned become more probable. When considered from the standpoint of deep learning, these types of processing-structure relationships can be seen as changes to the distribution of expected grain morphologies for a given material system. While learning just the standard class of polyhedrons might be a relatively simple task, additional processing possibilities present changes to the shape distribution that vary based on the functional constraints of the applied process. This type of constraint-based change in morphology is fundamentally similar to the learnable constraints for other functional object classes, and building off this conceptual relationship is the premise for this investigation.

Here, we develop a novel network architecture that learns the shape of volumetric data with functional constraints and applies it to the problem of grain morphology of polycrystalline metals. This approach integrates recent advances in Generative Adversarial Networks (GANs) ([3]) into the materials domain to generate new grain morphologies based on the distributions of training data. As grains are stochastic and valid morphologies are not always recognizable, network training and refinement was initially performed on functional objects (e.g. chair, guitar, etc.) from a publicly available dataset built for deep-learning-based object recognition, and the network was subsequently trained to

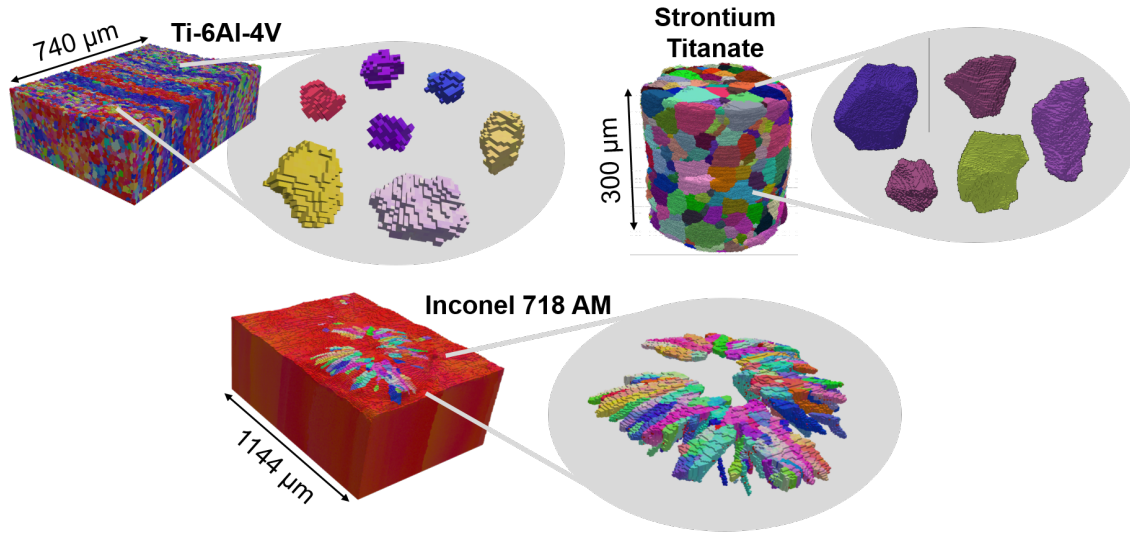


Figure 5.1: GRAIN DISTRIBUTIONS ACROSS MATERIALS AND PROCESSES: 3D datasets collected from Ti-6Al-4V (top left), strontium titanate (top right), and additively manufactured (AM) Inconel 718 superalloy (bottom) samples with detailed examples of specific grains contained within. Data shown was gathered with 3D electron backscatter diffraction (EBSD). Average grain size is much larger relative to voxel size resolution in the strontium titanate dataset compared the Ti-6Al-4V dataset, resulting in fewer total grains, but better defined grain facets. The Inconel 718 sample and the grain contained within the melt pool on the top surface of the sample are elongated due to the fast cooling rates present in AM processes.

generate Ti-6Al-4V microstructures. The resultant grain morphologies were evaluated for adherence to the shape constraints of actual polycrystalline metals. The results of this analysis are discussed in detail, and the limitations and potential benefits of this type of network-based approach are discussed in the context of the physics of crystalline materials.

5.2 Previous Studies in 3D Object Generation

As this investigation builds on the foundations of machine learning for 3D object recognition and synthesis, we will briefly describe relevant established approaches. Until recently, the majority of studies on 3D object generation have focused on the retrieval or

combination of components of the object of interest, such as in [110], [111], and [112]. In these approaches, for a given database of shapes, a probabilistic graphical model learns the geometric and semantic relationships that will yield a stylistically compatible object. Taking this a step further, [113] represented geometric 3D shapes as probability distributions of binary variables on a 3D voxel grid and was able to successfully demonstrate shape completion from “2.5-dimensional” depth maps. A related investigation from [114] proposed a network that used the ShapeNet dataset to learn a mapping from 2D images to their underlying 3D shapes, enabling the generation of a 3D representation of an object from an input 2D image. This led to efforts from [115–117] and [118] to generate 3D representations from 2D images. Although these methods showed encouraging results, most relied on some form of human supervision and did not focus on directly generating 3D objects from latent space, which can be thought of as a compressed data space used by the network to describe different possible output of interest. In this case, a lack of focus on latent space focus means these methods were not as well-suited for the recognition and synthesis of broadly described stochastic objects like grain morphologies. In the context of unsupervised 3D object synthesis, [119] proposed a promising autoencoder-based network to learn a deep embedding of object shapes, which yielded then state-of-the-art shape completion results, but overall, supervised approaches have historically had greater success with libraries of readily distinguishable objects.

When considering GAN-based approaches for shape generation, the most relevant approaches are from [4] and [120], which focus on 3D shape generation from a probabilistic latent space with feature-based learning and adversarial loss components. The 3D-GAN network from [4] generates 3D objects from a low-dimensional latent space, thus allowing for both the sampling of objects without a reference image or CAD model as well as the exploration of the 3D object manifold. Similarly, [120] proposed a novel 3D GAN network, but supplemented it with a 2D image enhancer network. This enhancer network was able

to effectively learn and feed image features into the 3D model generator to synthesize high quality 3D data. While both of these networks deliver on their promise to provide a solution to the 3D model generation problem, they offer somewhat limited resolution and detail in the shapes that they generate. Furthermore, enhancement of 3-dimensional object results using 2-dimensional supplemental images is not readily achievable with images of materials microstructure because most, if not all, 2D microstructure imaging techniques offer only a planar image of surface microstructure with minimal depth of field, no photographic perspective, and no real means by which underlying 3D microstructure can be inferred.

5.2.1 Materials-Specific Approaches

Within the materials domain, the earliest computational approaches used to generate microstructural morphologies derive from physics-based models, which have been recognized for both their high level of detail and realistic output. A study by [121] used a Voronoi tessellation model that simulated ceramic grain boundary evolution based on well-established equations developed by [122], [123], and [124]. Additionally, research approaches by [125] combined Monte Carlo simulations and grain growth kinetics to model metal crystallization. These physics-based models are promising, but often require significant computational power and detailed knowledge of the energetic of the system. These are not always readily available, especially in more complex processing scenarios. To avoid this knowledge and computational burden, models have also been developed to generate microstructures based primarily on their statistical properties. For example, [5] and [126] used statistical descriptors to generate microstructures using tessellation and ellipsoid coarsening, respectively. A method for the robust comparison of synthetic microstructures (e.g. generated by the model of [126]) with experimental results

is described in [127], with the finding that certain morphological descriptors resulted in improved discrimination, and that the starting shapes for grain generation impacted different descriptors. Approaches such as these are computationally efficient and versatile, compared to physics-based models, but tend to result in less realistic grain morphologies and limited accuracy in local grain environment descriptions.

Beyond physics-based models, machine learning has been employed to explore the material microstructure space of non-crystalline materials. [128] used graph neural networks of 3D atomic arrangements to describe general amorphous structures, such as those found in glassy materials. These graphical descriptions were then used as predictive tools to explore how the observed structure affects mobility and resultant glass properties. This approach showed promise for glasses, but does not extend to the crystalline domain. In the context of more general two-phase microstructures, [129] used 3D convolutional neural networks to characterize possible stochastic microstructures made from filtered noise. These types of stochastic microstructures have also been investigated in 2D by [130] using classification trees and by [131] using GANs.

Machine learning techniques have also been previously applied specifically to experimental crystalline structures. Investigations by [132] compiled a database of 2D ultra-high-carbon steel micrographs, and classified these images into microstructures based on the distributions of their microstructural features ([133]). In [134], a Wasserstein GAN with gradient penalty was used to generate 2D microstructures using the database of [132] as training data. Related to this investigation, [135] used StyleGAN to generate various 2D microstructures, and [136] used a convolutional deep belief network to generate 2D microstructures of the same titanium alloy explored in this study. Approaches such as these have strong experimental underpinnings; however, validation of microstructures is difficult in 2D. The challenge in 2D microstructure assessment stems from the fact that material micrographs are planar images of solid material that offer very little

or no perspective-based information from which 3D appearance can be inferred. For quantitative evaluation, the challenge becomes virtually impossible, as establishing 3D structure from 2D for anisotropic shape classes is mathematically intractable ([137]). In order for the output from machine learning-based models to be fully comparable to material microstructures, generation and evaluation robust 3D information on material microstructure.

Noteworthy advances have also been made in 3D crystalline material generation and evaluation in property space. [138] used structural optimization to explore the microstructure space of the iron-gallium alloy Galfenol. This approach focused on techniques for optimizing grain arrangement in the Galfenol microstructure to achieve an orientation distribution that would improve desired properties. While this approach allows for extensive exploration of property space, the basis of the model is theoretical, and there are no means of verifying that the proposed microstructures can be physically realized with available processing techniques. More recently, [139] used GAN-based approaches to generate three-phase microstructures of solid oxide fuel cells, which were then evaluated against experimental results and similar structures generated by the statistical methods of [126]. This approach evaluated microstructures both at the feature and property level, and demonstrated strong representation and promising results. As the focus of the network was to generate distributions of the three phases present in the fuel cell anode, namely yttria-stabilized zirconia, nickel, and pore/void, it was tailored around producing phase distributions specifically and did not address features within those phases. Often, deep learning approaches rely on some form of experimental ground truth data, and for the investigation presented here, the experimental ground truth was gathered using a technique known as electron backscatter diffraction (EBSD). EBSD is a scanning electron microscopy technique where electrons from the microscope beam are diffracted by a crystalline structure according to Bragg's law ([140]), forming a pattern that can be

indexed to determine information about the underlying structure. This approach is a well established crystallography technique whose capabilities are discussed in [141] and [142]. Typically, the first step in processing EBSD data is indexing, where the diffraction pattern gathered from the microscope is mapped to a particular crystal structure and orientation. The most conventional method of indexing is mapping using a Hough transform, but other more sophisticated mathematical and machine-learning-based approaches, such as those presented by [21, 143, 144] and [55], allow for more efficient and higher quality indexing with reduced error. Because indexing is a mapping of a diffraction patterns to specific locations, all EBSD data is inherently in pixel or voxel form. After the data is indexed, additional material information can be extracted based on the relative orientations and deformation states of the mapped crystalline grain structures. Because this study is interested in grain morphology, all ground truth data used in this study is fully indexed and segmented into grains based on continuity of crystalline domains. However, in the investigation presented here, much of the novelty arises not from the demonstrated experimental application, but rather from the extensibility of the approach. By learning morphologies from general distributions of relevant features, a network can be produced that is applicable to a wide variety of materials systems and applications with no needed architectural refinement.

5.3 Method

In this investigation, we demonstrate the GAN-based recognition and synthesis of crystalline grain morphologies using learned feature distributions built on the same fundamental principles used for the generation of other functional objects. We start with readily available databases of easily recognized functional objects to refine the GAN's synthesis capabilities and evaluate network performance on recognizable objects. We then

apply the established architecture to materials systems, more specifically grain morphologies, where the scope of accessible data is limited and recognition is not straightforward. While this is not directly equivalent to transfer learning, it demonstrates how available information databases might be used to teach relevant broader concepts to networks, which, in the right context, can be applied to scientific problems where large-scale datasets are not easily obtained.

5.3.1 Architecture and Training

First proposed by [3], a simple GAN consists of a generator G and a discriminator D . The generator tries to synthesize samples that look like the training data, while the discriminator tries to determine whether a given sample is a real sample that originated from the ground truth data distribution or is synthesized from the generator. The discriminator D then outputs a confidence value $D(x)$ of whether input x is real or synthetic. A basic layout of the GAN architecture is shown in Figure 5.2. StyleGAN, developed by [145], garnered widespread attention for its life-like image quality and unsupervised high-level attribute separation in the generated output. Because of these characteristics, we used StyleGAN as the base architecture for our network. Instead of passing a random noise vector z to the generator, z is first mapped to an intermediate latent space W , which is transformed into spatially invariant styles $\mathbf{y} = (\mathbf{y}_s, \mathbf{y}_b)$. This is then used to control the generator through adaptive instance normalization (AdaIN) at each convolutional layer. The AdaIN is described by [145] using the form:

$$\text{AdaIN}(\mathbf{x}_i, \mathbf{y}) = \mathbf{y}_{s,i} \frac{\mathbf{x}_i - \mu(\mathbf{x}_i)}{\sigma(\mathbf{x}_i)} + \mathbf{y}_{b,i} \quad (5.1)$$

where \mathbf{x}_i is a feature map normalized separately for each instance i and then scaled and

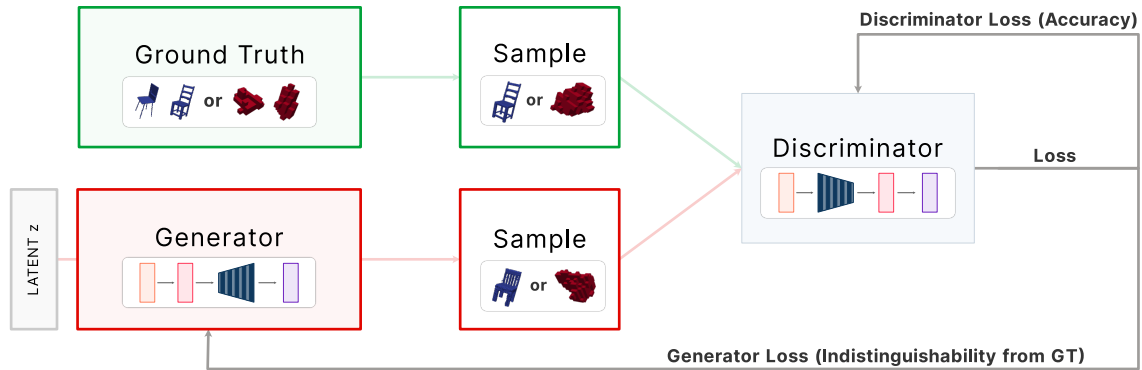


Figure 5.2: GENERATIVE ADVERSARIAL NETWORK (GAN) ARCHITECTURE OVERVIEW. A layout of the architecture of a traditional GAN, as described by [3].

biased using the corresponding scalar components \mathbf{y}_s and \mathbf{y}_b from style \mathbf{y} . Mapping to intermediate latent space allows for disentanglement, which provides clearer tracking of the influence of latent space variations on the generated output. For generative architectures, latent space is the random source noise used as input, and disentanglement, as described by [145], allows for the fluctuations in this space to be separated and applied independently for different generated features. This helps produce better results in scenarios where features are not represented proportionally to the initial sampling, such as for the anisotropic feature distributions expected for both the functional objects and experimental material data investigated here. Additionally, the use of style based representation allows for better evaluation of network understanding at different feature levels, which is particularly useful in stochastic structures of variable size, such as grains.

5.3.2 M-GAN Network

In our 3D Generative Adversarial Network (M-GAN), an initial latent space vector z of size 512 is chosen through Gaussian sampling of latent space, similar to the work of [145]. In their work, [145] determined that a size of 512 for z was sufficiently large to keep all components of latent space disentangled. The generator G then maps this latent

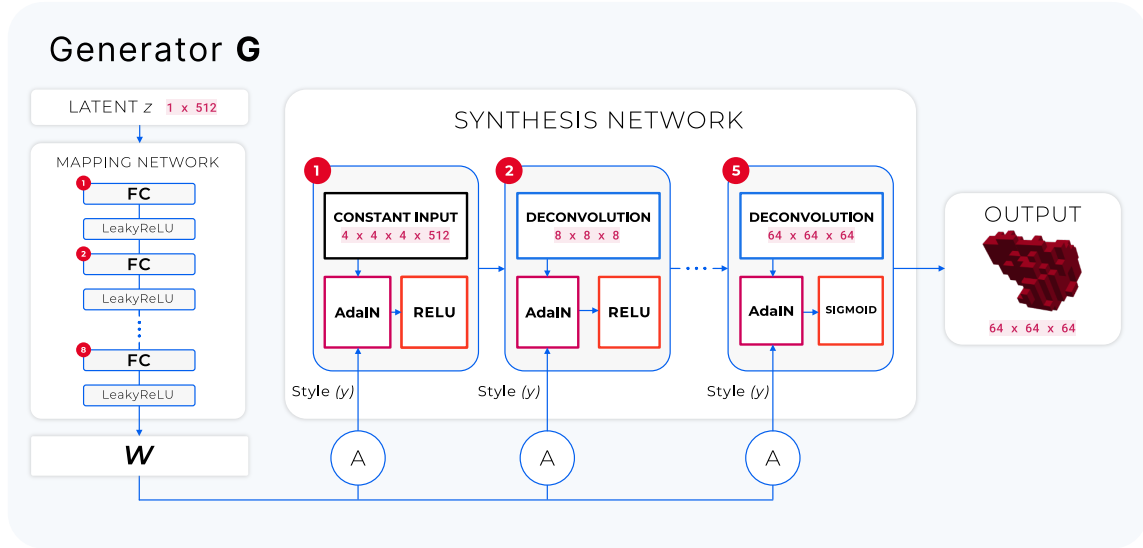


Figure 5.3: GENERATOR ARCHITECTURE OF M-GAN. A mapping network comprised of eight Fully Connected layers (FC) with Leaky ReLU activation function after each FC layer takes as input a 512 dimensional latent vector z . The output is then mapped to an intermediate latent space \mathcal{W} , converted into styles using a learned affine transformation (A), and passed through an AdaIN operation for each of the five blocks in the synthesis network. Block 1 passes constant input through AdaIN and ReLU activation functions, while Blocks 2-5 are deconvolution blocks progressively grown from $8 \times 8 \times 8 \rightarrow 64 \times 64 \times 64$.

vector into an intermediate latent space \mathcal{W} using a mapping network of 8 fully connected layers with a LeakyReLU activation function after each layer. The output \mathcal{W} is then converted into the styles previously described as \mathbf{y} using a learned affine transformation, denoted as A in Figure 5.3, and these styles are used to control adaptive instance normalization (AdaIN) in the synthesis network, as described in Equation 5.1. This localization of styles better preserves the small scale physical features of each object or grain, allowing for more detailed morphology output compared to other networks ([4, 120]). Our synthesis architecture diverges from that of StyleGAN by using blocks consisting of a 3D deconvolution passed through an AdaIN operation and ReLU activation function. The synthesis network has a total of five blocks, with the first having a constant input vector and normalization, and the fifth having a sigmoid activation function instead of

a ReLU. In comparison, StyleGAN blocks use upsampling followed by two alternating convolution layers and AdaIN operations, with noise introduced after each convolution to add stochastic variation. However, in 3D, this noise was found to add instability during training, and was therefore not included in the M-GAN network. Architecturally, the discriminator is similar to that of [4] in that it consists of 5 progressive 3D convolution layers with no downsampling. Each of these convolution layers has a LeakyReLU activation function except for the last layer, which has no activation function. The first four layers have a kernel size of 4, stride size of 2, and padding size of 1, and the last block has kernel size of 4 and stride size 2 with no padding. The output of this last block goes directly into the loss function. In this investigation discriminator makes its decisions (real vs. generated) based on collections of multiple objects from the same class, rather than just singular instances, similar to the method presented by [146].

5.3.3 Material Dataset for Training

The data used to train M-GAN includes of a combination of object based data for network refinement and evaluation as well as fully-indexed EBSD data for application to material microstructures. For object-based data, six major categories from the ShapeNet Dataset by [147] were used in the network training: car, chair, plane, guitar, sofa, and rifle. Each category has a 128-sample training set. All objects are presented in a voxel-based format, and each individual object instance is contained within of $64 \times 64 \times 64$ voxel volume.

The material microstructure data used in this study was 3D EBSD gathered by J. Wendorf, A. Polonsky, J. C. Stinville, and M. P. Echlin ([1]) using a method developed by [148] known as the Tribeam system, which performs rapid serial sectioning using ablation with ultrashort pulse femtosecond lasers. The material investigated here is a

polycrystalline wrought titanium alloy containing 6.75 wt.% aluminum and 4.5 wt.% vanadium. This alloy, commonly referred to as Ti-6Al-4V, has applications in turbine engines, aerospace, and medical prostheses. The grain information is gathered at the voxel level using electron backscatter diffraction, with each voxel having a size of $1.5 \mu\text{m} \times 1.5 \mu\text{m} \times 1.5 \mu\text{m}$. Each grain is passed to the network inside a cubic volume of size $64 \times 64 \times 64$ voxels.

The full dataset is illustrated in Figure 5.4. This Ti-6Al-4V dataset was originally gathered to study slip behavior and long-range plastic deformation across multiple length scales ([149]) as well as explore how microtextured regions in titanium alloys affect overall mechanical response ([1, 150]). To better capture long-range mechanical response, the sample size is relatively large compared to the grain size, so many grains are relatively small compared to the voxel resolution and detailed grain facets are not easily resolved. Despite these resolution limitations, this set has the major advantage of offering a large number of grains as a good foundation for network training. In total, 84,215 grains from this dataset are used for network training.

Stable training of the GAN was achieved by setting the learning rate for both the generator and discriminator at 0.0002, with a batch size of 16. Similar to [4], the discriminator in this approach is updated five times for each generator update and employs the Adam optimizer described by [151] with $\beta = 0.5$. The output of the network is in a $64 \times 64 \times 64$ dimensional space. The Wasserstein loss as described in [152] with a gradient penalty is used for the discriminator and the generator, which are defined as follows:

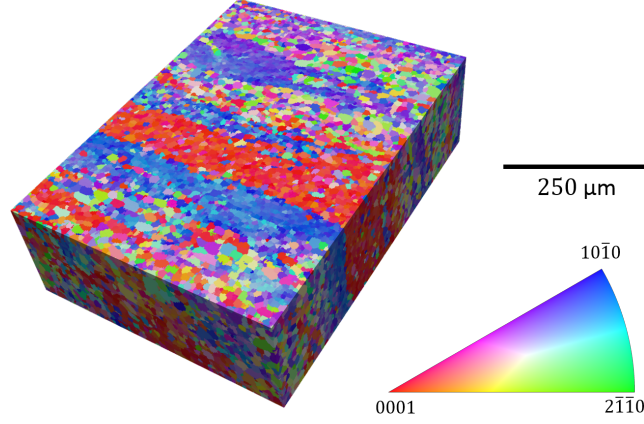


Figure 5.4: EXPERIMENTAL TRAINING SET: Ti-6Al-4V dataset, whose grains were used to train M-GAN on grain shape recognition. Sample is shown to scale in inverse pole figure (IPF) color. The IPF color indicates the orientation for each individual voxel, and these orientations are used in the segmentation process to identify individual grains, which are subsequently used for training.

$$L_{\text{D-Loss}} = -\frac{1}{m} \sum_{i=1}^m D_{\phi}(x^{(i)}) + \frac{1}{m} \sum_{i=1}^m D_{\phi}(G_{\theta}(z^{(i)})) + \frac{1}{m} \sum_{i=1}^m \lambda (\|\nabla_{\hat{x}^{(i)}} D_{\phi}(\hat{x}^{(i)})\|_2 - 1)^2 \quad (5.2)$$

$$L_{\text{G-Loss}} = -\frac{1}{m} \sum_{i=1}^m D_{\phi}(G_{\theta}(z^{(i)})) \quad (5.3)$$

Here, G_{θ} is the generator network, D_{ϕ} is the discriminator network, $z_{i=1}^{(i)m} \sim p(z)$ is a batch of random noise from latent space ($p(z)$ = population distribution for latent space), $x^{(i)}$ is instance i of real data from a batch of size m such that $x_{i=1}^{(i)m} \sim \mathbb{P}_r$ (\mathbb{P}_r = population distribution of real data), ∇ is the gradient operator, λ is a gradient penalty coefficient as described by [153], and $\hat{x}^{(i)}$ is defined as $\epsilon x^{(i)} + (1 - \epsilon)G_{\theta}(z^{(i)})$ where ϵ is a uniform random variable in $[0,1]$. This Wasserstein loss metric, also known as the "earth-mover" metric, quantitatively compares the difference between two distributions by considering the area

under each distribution curve and measuring how much area would have to be moved from one distribution for it to match the other. When applied to feature distributions, it is a measure of how well the features in the generated data match those found within the feature distribution of the ground truth dataset. This type of comparison uses the ground truth data as guide for training without relying directly on one-to-one mappings between generated and ground truth data, which enables the network to better produce results that are similar to the ground truth without being direct copies. By definition, this lack of a one-to-one mapping also means this approach is considered an unsupervised learning method. Furthermore, since this approach is unsupervised, all hyperparameters used in this study are preset before training, and all validation of data is done as analysis on objects generated by the network during inference time.

5.3.4 Evaluation

We evaluate our network across several areas. First, we show qualitative results of generated 3D objects from the ShapeNet dataset by [147]. Then, we evaluate the unsupervised learned representations from the discriminator by using them as features for 3D object classification. Next, quantitative results are shown on the popular benchmark ModelNet dataset from [113]. Finally, we apply the network approach to grain morphologies from real, experimentally gathered materials data. Because grain shapes are stochastic, direct recognition of resultant objects is not straightforward, so shape quality must be validated by other means. Here, shapes are validated using the moment invariant approach described in [154].

We train one M-GAN for each object category using a 512 dimensional random vector that follows a normal distribution with mean 0 and variance 0.2. We compare our generated objects with [4], because [120] used an enhancer network and additional information

from 2D rendered images during training. Our network synthesizes high-resolution 3D objects ($64 \times 64 \times 64$) with detailed shape information trained from only 3D input. To ensure that the network is not simply memorizing the training data, we analyze synthesized objects using comparisons to nearest neighbors in discriminator feature space as described by [4]. For this analysis, discriminator feature vectors are captured for both the generated samples and ground truth data, and nearest neighbors in feature space are determined using minimum ℓ_2 distance, which is far more efficient than direct 3D object comparison. Feature analysis shows that the generated samples were not identical to their nearest neighbors.

To evaluate the unsupervised learned features from our network and assess performance, we use the feature extraction approach discussed in [4] to provide a means of comparison, as there is no established standard. In this approach, features identified in the convolution layers of the discriminator are collected for a variety of different categorically classified objects. This library of features is then integrated into a support vector machine whose classification performance is tested using a new dataset of equivalently classified objects. We train our M-GAN network on six object categories (bed, car, chair, plane, sofa, table) from [4]. Each object category has 25 samples, similar to [4] in the training set from the ShapeNet dataset.

Next, we use the ModelNet dataset from [147] to evaluate the unsupervised features learned by our network. The ModelNet dataset has two categories: ModelNet10 (10 classes) and ModelNet40 (40 classes). ModelNet10 has a total of 3991 training samples and 908 test samples. From this dataset, we use 100 samples from each category, totaling 1000 training samples. ModelNet40 has a total of 9843 training samples and 2468 test samples. From this dataset, we use up to 100 samples from each category when available, totaling 3906 training samples, as some categories had fewer than 100 samples. For ModelNet10 and ModelNet40, we use the entire test dataset provided, though fewer

training samples than were used by [4] and [120].

When evaluating feature extraction accuracy, to provide a fair comparison, we use the same kernel size = {8, 4, 2} defined by both [4] and [120], as well as the defined stride = {4, 2, 1} by [120]. We calculate features from the second, third and fourth layers of the trained discriminator, which are then concatenated after applying max-pooling with the defined kernel size and stride. We then train a linear Support Vector Machine (SVM) on training features and calculate classification accuracy on the test features.

Functional object data from the ShapeNet and ModelNet sets are well suited for network refinement and performance assessment, in part due to their easy visual recognition. Grain shapes, on the other hand, have a stochastic morphological appearance, so they cannot be easily evaluated by visual means or by direct object comparison. For this reason, we use statistical distribution comparison of 3D moment invariants to assess the quality of generated results for single grains. Much like the image moments used in 2D analysis, 3D moment invariants are integration-based descriptors that numerically quantify an object based on the distribution of its solid volume. These types of invariants have been used previously by [155] as shape descriptors for general 3D objects, and in materials science by [154] and [156] to describe the shapes of particles such as grains and inclusions. Following the approach of [154], Cartesian moment descriptors μ_{pqr} take the form:

$$\mu_{pqr} = \int \int \int x^p y^q z^r F(r) dr \quad (5.4)$$

where $F(r)$ is a characteristic function that has a value of one in material regions and a value of zero in void regions. As all ground-truth data is in voxel-based form in this investigation, all integrations herein are calculated as Riemann sums in voxel space rather than as continuous integrals. Following this Cartesian moment form, μ_{000} is directly equal

to the volume V of an object, and the centroid (X_c, Y_c, Z_c) of an object can be expressed in the following form:

$$(X_c, Y_c, Z_c) = \left(\frac{\mu_{100}}{V}, \frac{\mu_{010}}{V}, \frac{\mu_{001}}{V} \right) \quad (5.5)$$

Working from a coordinate space originating at the centroid, the non-normalized moment invariants $\mathcal{O}_1, \mathcal{O}_2, \mathcal{O}_3$ are as follows:

$$\mathcal{O}_1 = \mu_{200} + \mu_{020} + \mu_{002} \quad (5.6)$$

$$\mathcal{O}_2 = \mu_{200}\mu_{020} + \mu_{200}\mu_{002} + \mu_{020}\mu_{002} - \mu_{110}^2 - \mu_{101}^2 - \mu_{011}^2 \quad (5.7)$$

$$\mathcal{O}_3 = \mu_{200}\mu_{020}\mu_{002} + 2\mu_{110}\mu_{101}\mu_{011} - \mu_{200}\mu_{011}^2 - \mu_{020}\mu_{101}^2 - \mu_{002}\mu_{110}^2 \quad (5.8)$$

These can be normalized to object volume to produce the moment invariants $(\Omega_1, \Omega_2, \Omega_3)$:

$$(\Omega_1, \Omega_2, \Omega_3) = \left(\frac{3V^{5/3}}{\mathcal{O}_1}, \frac{3V^{10/3}}{\mathcal{O}_2}, \frac{V^5}{\mathcal{O}_3} \right) \quad (5.9)$$

The distributions of these moment invariants are used to evaluate the shape quality of the generated grains. Additional analysis of 3D moment invariants can be found in [154].

5.4 Results

5.4.1 Object-Based Data

For object-based datasets, the network showed strong performance. Examples of generated object output based on ShapeNet training are shown in Figure 5.5, and network performance against existing architectures is shown in the Table 5.1. With fewer training samples than [4] and [120], we achieved 92.29% accuracy on the ModelNet10 dataset and

85.08% accuracy on the ModelNet40 dataset. If we use a comparable training set size to [4], we achieve a 2.29% improvement on ModelNet10 and a 3.78% improvement on ModelNet40 dataset. Furthermore, [120] uses all available training samples as well as additional rendered 2D images for both datasets, compared to our use of fewer training samples and only 3D input.

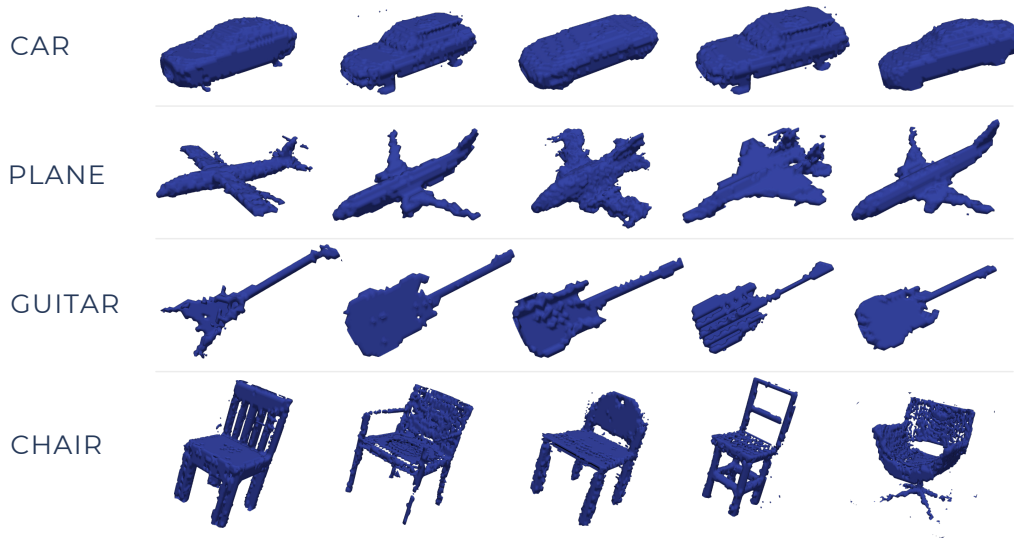


Figure 5.5: SHAPENET OUTPUT. The M-GAN network generates detailed shape information for a diverse range of 3D objects. As the network learns from shape feature distributions, these objects are similar but not identical; such as the different styles of chairs shown in the bottom row. The variation within a class of generated objects that satisfies functional requirements is applied to the generation of grains across different material and processing classes as shown in Figure 5.1.

Since all benchmarking for M-GAN was done on recognizable objects with well-defined orientations, no rotational augmentation of datasets was used during training with objects or grains, and all results are shown in their as-generated orientation. Analysis of generated objects indicates that the M-GAN network is not generating any rotated objects in cases where rotation effects are distinguishable. For microstructural objects like grains, since they are fundamentally stochastic and have no defined top or bottom, rotation augmentation may prove beneficial as an additional training tool, especially for anisotropic grain structures, but the influence of rotational augmentation is not explored

Method (Supervised)	ModelNet10 (%)	ModelNet40 (%)
3D ShapeNets([113])	83.54	77.32
VoxNet ([157])	92.00	83.00
Geometry Image ([158])	88.40	83.90
PointNet ([159])	77.60	-
GIFT ([160])	92.35	83.10
FusionNet ([161])	93.11	90.80
Method (Unsupervised)	”	”
SPH ([162])	79.79	68.23
LFD ([163])	79.87	75.47
VConv-DAE ([119])	80.50	75.50
3D-GAN ([4])	91.00	83.30
3D-GAN ([4]) (\approx 100 samples)	90.00	81.30
Ours (M-GAN) (100 samples)	92.29	85.08

Table 5.1: Comparison of object-based performance for the ModelNet10 and ModelNet40 datasets using the feature extraction approach described by [4].

in this work.

5.4.2 Material Data

Following functional object generation, the network was trained on experimentally gathered Ti-6Al-4V grain morphologies. Representative volumes of grains generated using the M-GAN are shown in Figure 5.6. Unlike the benchmark results, which are shown as contour displays, these figures are shown in voxel form, which matches the representation of the ground-truth data. It should be noted that some generation artifacts were present in $64 \times 64 \times 64$ volume containing these grains, with the most common of these artifacts being filled single voxels at random locations in the render volume. Some examples of these artifacts can be seen in the contour images of the benchmark set shown in Figure 5.5. In the case of grain volumes shown in Figure 5.6, only the largest connected volume is shown. These representative volumes show shapes that would be reasonably expected of grains in an equiaxed polycrystalline metal like wrought Ti-6Al-4V, but these

representative volumes alone are insufficient verification of generated grain quality.

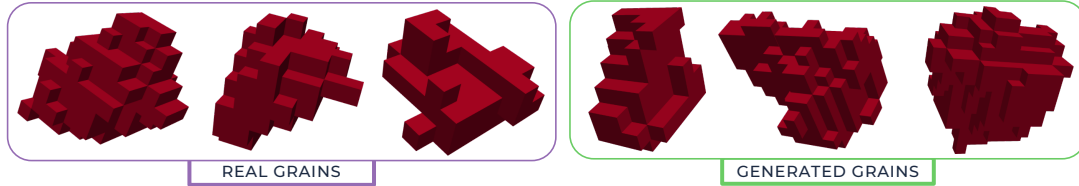


Figure 5.6: VISUAL COMPARISON OF GRAINS. Sampling of grains from real and generated sets are shown to verify visual similarity. Voxel size is $1.5 \mu\text{m} \times 1.5 \mu\text{m} \times 1.5 \mu\text{m}$. For generated grains, only largest connected component is shown. Stochastic nature of grains means no directly matching shapes are expected. Shape distributions are compared in Figure 5.7.

For quantitative shape evaluation, the distributions of the 3D moment invariants $(\Omega_1, \Omega_2, \Omega_3)$ evaluated in this study are shown in Figure 5.7. The ground truth dataset contained 84,215 grains and the M-GAN generated a total of 150,000 grains. For each of the three invariants, less than 0.2% of the grains had values that were either infinite or nonphysical across both the ground truth and generated sets. In observed cases, this resulted from very small grain structures that were linear or planar in nature, which led to extremely large errors in Riemann summation during the calculation of the invariants. Due to their nonphysicality, these values were omitted from the statistical distribution comparisons.

A comparison of the experimental moment invariant distributions in this investigation compared to those analyzed in [127] shows that ones presented here are of slightly lower average value, which indicates grains being more cube-like in shape. This is expected though, given that the grains in this set are of lower resolution in order to capture a large volume, and lower resolution will mean more cube-like grain features. The means and standard deviations for each of the distributions are shown in the insets of Figure 5.7. These values show that the generated data consistently exhibited lower means and larger standard deviations. Comparison of the distribution shapes in Figure 5.7 reveals the cause to be that the generated data has a range of moment invariants values in the

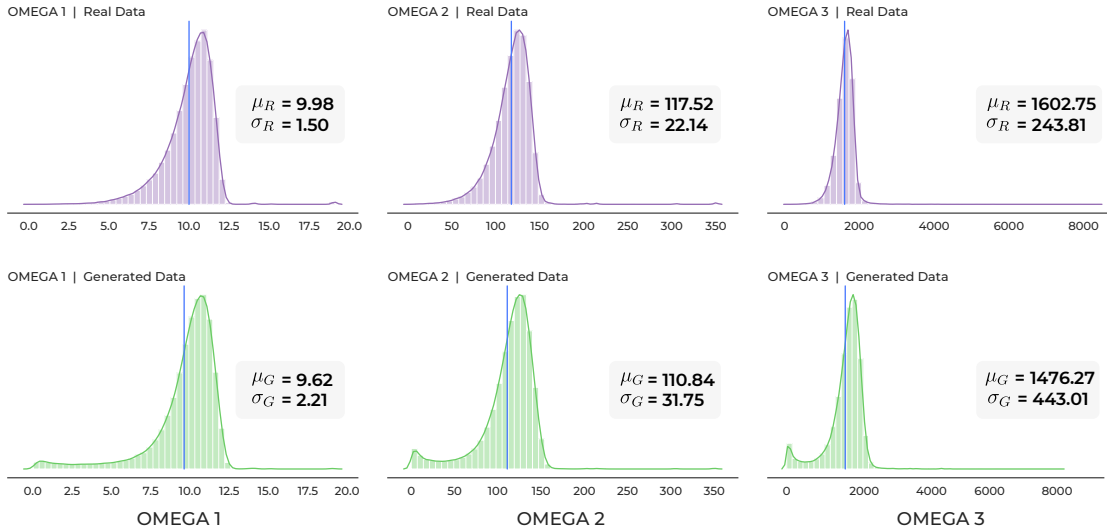


Figure 5.7: QUANTITATIVE EVALUATION OF M-GAN GRAIN GENERATION: Moment invariants ($\Omega_1, \Omega_2, \Omega_3$) are used to evaluate the similarity of real and generated shape distributions, as grains shapes are stochastic and correctness is a matter of shape probability rather than object recognition. Histograms are shown for both real (top) and generated (bottom) grain shapes. Vertical lines indicate the mean.

region close to zero that is not seen in the real dataset.

Analysis of generated grains from these lower magnitude moment invariant values revealed two common issues. The first is that many of the generated grains in this region are smaller in volume than the ground truth. In the ground truth dataset, the smallest grains that could be reliably distinguished from artifacts and noise are on the order of 25 voxels in volume, but many of the generated grains with low moment invariant values had volumes of fewer than 20 voxels. While small volume alone is not necessarily a concern, these smaller grains had a greater negative interaction with the other major issue of generator noise, which was also noticed in the some object based data as in Figure 5.5. Although the generator noise is not displayed in Figure 5.6 for visual clarity, this noise is still included in the moment invariant calculation both to minimize quantitative bias and because no steps are included in the architecture or training process to directly mitigate noise. Noise removal with the aid of domain knowledge is a widely accepted practice

for microstructural data, but since the goal of this work is direct evaluation of network performance, denoising is considered a post-processing step that is outside the scope of the network. This consideration of noise is critical because some generated grains had very small volumes compared to the ground truth, and many more still had relatively small volumes compared to the $64 \times 64 \times 64$ voxel region in which they were generated. Thus, generator artifacts located far away from the grain itself can cause significant shifts in the centroid location used to establish the coordinate basis for the calculation of the moment invariants, thereby creating distortions in the data. However, beyond these factors, the overall shape and position of the main peaks for both distributions are very similar, indicating that the M-GAN network is producing data that is similar to the ground truth data without directly replicating it.

5.5 M-GAN BisQue Module

We have successfully developed and implemented a BisQue module for 3D grain generation, as illustrated in the Figure 5.8. Within the BisQue user interface, users can simply click the 'Run' button, and the module will efficiently generate single 3D grains from the latent space. This streamlined process on the BisQue platform offers a user-friendly and efficient approach to 3D grain generation, making it accessible for users.

5.6 Discussion

5.6.1 Relation to Material Physics

While a quantitative analysis of morphology shows promising results, network performance must be considered in the context of actual material physics. Here, the issue of small grain volumes brings to light another challenge with GAN-based morphology syn-

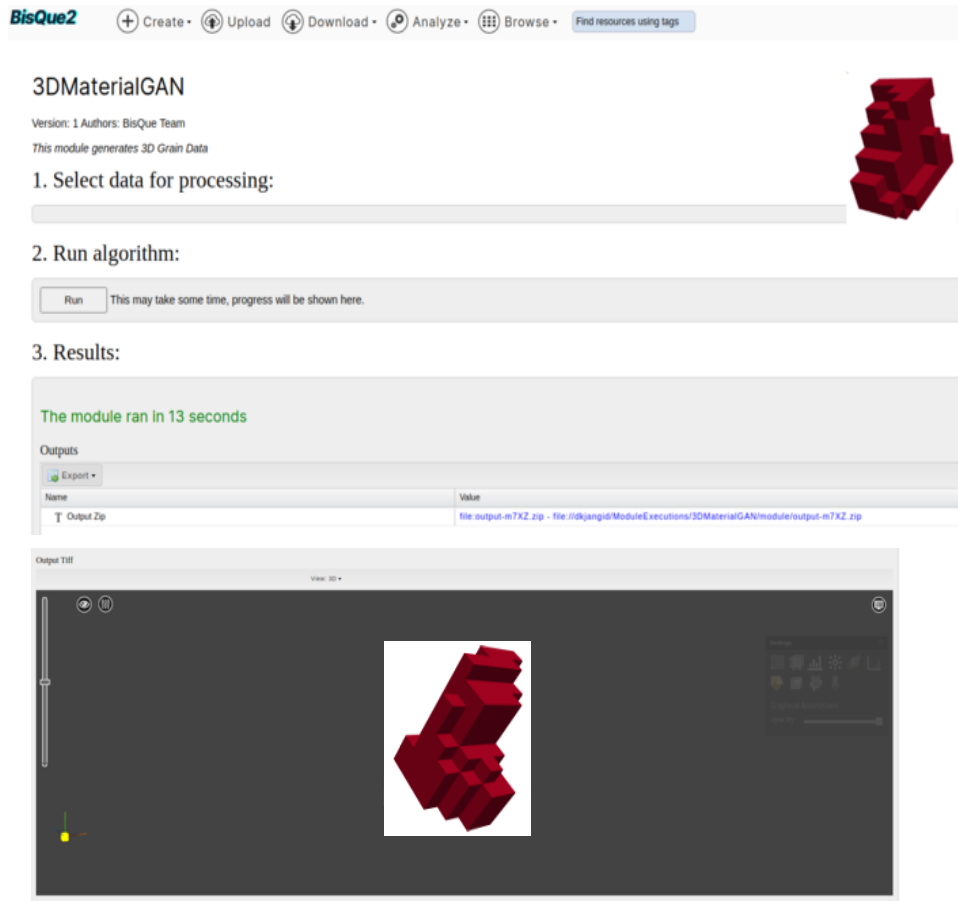


Figure 5.8: BISQUE MODULE FOR MGAN. This module generates a 3D single grain from a random noise vector.

thesis, namely that of size-shape relationships. Throughout object training, the network is constrained to a $64 \times 64 \times 64$ voxel final output size, regardless of the object morphology being generated. As such, it learns to recognize morphological features independent of the object size, and voxels are assigned whatever physical length scale is necessary to render the object in the available volume. Thus, when considering for example, cars and guitars, the relevant features on both of these objects are learned in the context of the available rendering scale, even though the physical size of the guitar and car are different.

Grain Size Issues: In many scenarios, size-independent morphological recognition is an asset that offers greater versatility. However, for materials data, it has some unde-

sired consequences. In the previously discussed case of low-volume grains, the network generated objects smaller than the experimental resolution of the dataset, even going as far as to produce an instance of a grain with single-voxel volume. While these results are inaccurate with respect to the ground truth data, this is primarily due to voxel-based resolution limitations rather than physical inaccuracies. Even for the most extreme case of a single voxel grain, if we apply the ground truth length scale of $1.5 \mu\text{m} \times 1.5 \mu\text{m} \times 1.5 \mu\text{m}$ per voxel, it is certainly possible for a grain with a diameter of $1.5 \mu\text{m}$ to exist in a variety of real materials. However, regardless of material, that grain will possess morphological constraints that are not represented by a single-voxel rendering. This tells us the network has challenges with distinguishing feature constraints associated with physics from those associated with resolution limitations. This is a challenging problem to address in EBSD images, where data is gathered using a rastering electron beam, and therefore will always be pixel- or voxel-based. Currently, the most promising outlook for resolving this issue is an automated means of enhancing 3D images with resolution artifacts, such as super-resolution, a survey of which is presented by [164]. With sufficient resolution enhancement, small-scale grains could be presented to the network at a volume scale where physical features can be readily recognized beyond resolution limitations. It may be possible to remove some of these resolution limitations using either networks that produce non-voxel output, or networks that allow for continuous resolution scaling, such as Infinity GAN by [165]. However, achieving meaningful results on materials data with either of these approaches would require detailed 3D information across a variety of different length scales, which is nontrivial to obtain.

Grain Shape and Connectivity Issues: The other challenge that must be addressed for crystalline materials is grain connectivity relationships. For both the object and material data, all learning was performed with independent renderings of single

objects. Recognition of individual objects regardless of context can be beneficial; however, in polycrystals, the shape of each grain depends on physical interaction with its surrounding neighbors. The network presented here has a good understanding of grain morphology, but cannot infer how different grain shapes are related to one another in the bulk material. It is possible that graph or label-based approaches may offer a means to clarify these 3D connectivity relationships, but to date, the available resources for achieving this are limited.

The presented M-GAN approach demonstrates how fundamental materials concepts, such as grain morphology, can be learned by network-based approaches in cases where an abundance of data is not readily available. Grain morphology is addressed herein as it is a principal component in understanding the relationships between microstructure and material properties. Network-based approaches such as M-GAN enable the generation of material features (e.g. grain morphologies) in a generalizable context. This broad generative capacity expands our exploration of the processing-structure and structure-property relationships that are critical to materials discovery and development. For example, the M-GAN approach is directly applicable to the generation of bulk synthetic polycrystalline microstructures from constituent components. M-GAN can readily be used to generate collections of grain morphologies based on a distribution of interest, and these morphologies can be iteratively packed and assigned crystallographic orientations to form a bulk polycrystalline solid. Network-based approaches like M-GAN also serve as a foundation to explore morphology-dependent relationships, such as the formation of micro-textured regions in titanium alloys ([166–168]), or the formation of thermomechanically-dependent grain structures, such as columnar grains from directional melt solidification in additive manufactured metals ([169–171]).

5.7 Conclusions

In this chapter, we present M-GAN, a generative adversarial network to recognize and synthesize 3D grain morphologies in crystalline microstructures. All objects are generated from latent space vectors without any supplemental 2D input.

Network capabilities are demonstrated on the ModelNet benchmark datasets and on an experimental 3D material dataset of Ti-6Al-4V. On benchmark datasets, the M-GAN yielded more reliable discriminator classification with exposure to fewer objects, highlighting the quality of its discriminator feature recognition. On experimental data, the M-GAN network produced results that were morphologically similar to ground truth without being a direct replication. The effects of generator noise produced some variation in the moment invariant distribution compared to experimental data, particularly at low value invariants, and most often with grains of smaller volume. These challenges with smaller grains indicate a limitation of the network in distinguishing grain shape features from spatial resolution artifacts.

This approach lays a foundation for the use of network based approaches to understand key relationships between grain structure and properties of crystalline materials. In future work, this approach could also be extended to relate grain shape to neighbor connectivity in the bulk material structure.

5.8 Appendix

5.8.1 Dataset Preprocessing and Postprocessing

Preprocessing: 3D objects in ShapeNet Dataset are in binvox format, while objects in ModelNet Dataset are in `.off` object format, which is converted into binvox format using the software found at: <https://www.patrickmin.com/binvox/>.

Postprocessing: The generated 3D objects from the trained network are produced in both vtk and binvox format. ParaView software has been used to visualize vtk files using the contour filter for shapes, and the threshold filter for grains. Binvox files are viewed using the ViewVox software (<https://www.patrickmin.com/viewvox/>).

5.8.2 Network Architecture Details

Generator Network: The Generator network is composed two parts: the Mapping network and Synthesis network.

Mapping Network: Mapping network has 8 fully connected layers of dimension (512, 512). Each layer has Leaky ReLU activation function with leak value of 0.2. The dimension of z , which is input to the mapping network, is (1, 512).

Synthesis Network: The synthesis network is composed of 5 styled convolutional blocks. The first block has a constant vector of dimension ($512 \times 4 \times 4 \times 4$), AdaIN normalization, and ReLU activation function. The second through fourth blocks have 3D deconvolutions with kernel size 4, stride of 2, padding size of 1, 3D batch normalization, AdaIn normalization and ReLU activation functions. The fifth block has a 3D deconvolution of kernel size 4, stride size of 2, AdaIN normalization, and Sigmoid activation function. The output dimension of the fifth layer in the Synthesis network is $64 \times 64 \times 64$.

Discriminator Network: The discriminator network has five convolution blocks in total. The first four have a kernel size of 4, stride size of 2, and padding size of 1, with LeakyReLU activation function of leak value 0.2. The last convolutional block has kernel size of 4 and stride size 2 with no padding and no activation function.



Figure 5.9: Generated results for Rifle

5.8.3 3D Object Classification for evaluation

For each object, we take the output of the second, third and fourth convolution layers of the discriminator and apply max pooling of kernel sizes of $\{8, 4, 2\}$ and strides $\{4, 2, 1\}$ respectively. We then concatenate the outputs into a vector, which is used by a linear SVM for training and classification. We use a Linear SVM with penalty ℓ_2 , squared hinge loss, and $C = 1$

While our network was able to learn the shape of a sofa successfully, it struggled to learn the attributes of a rifle. As shown in Figure 5.9, the shape of a variety of rifles are shown, but details such as the barrel, sight, trigger guard, and the magazine are sparsely captured.

Nearest Neighbor Test: To compute the nearest neighbor of a generated object, we calculate features of the object and all training samples by taking the output of fourth layer of the discriminator after applying max pooling with kernel size of 4. We then calculate ℓ_2 distance between features of a generated object and all training samples and find minimum distance. This is done in feature space because computing ℓ_2 distance directly between 3D objects is computationally expensive.

GENERATED

NEAREST NEIGHBOR

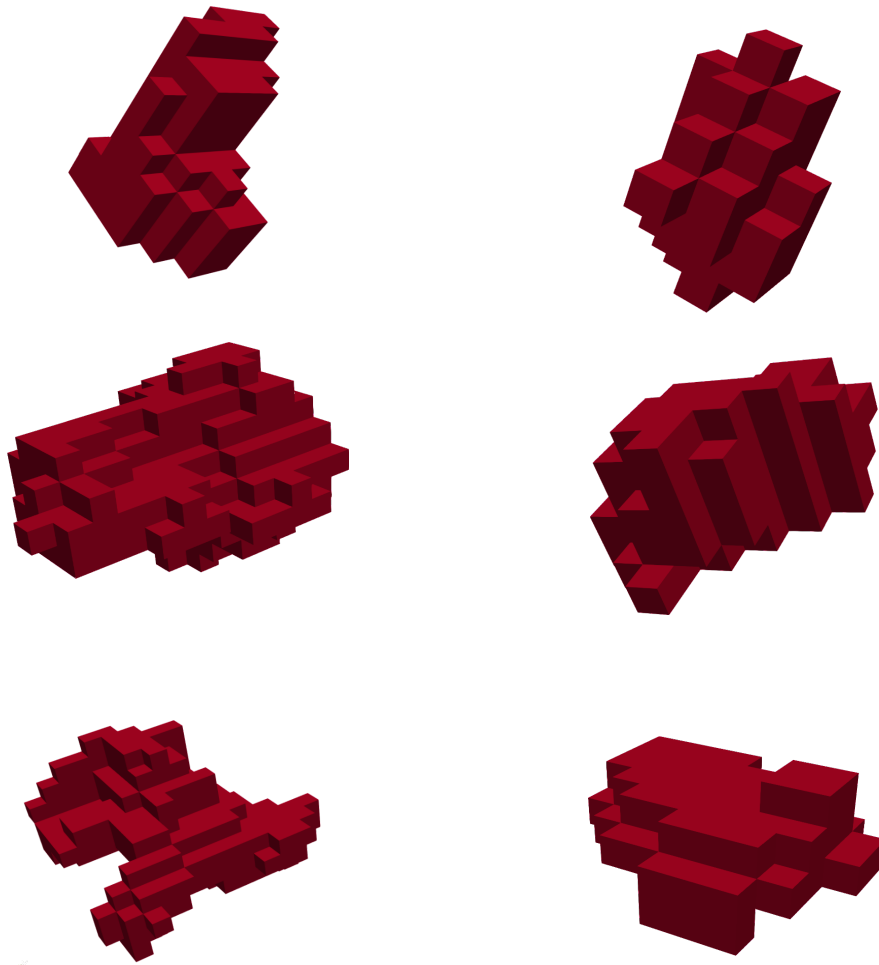


Figure 5.10: SINGLE GRAINS. Nearest Neighbor Results

GENERATED

NEAREST NEIGHBOR



Figure 5.11: SHAPENET DATASET. Nearest Neighbor Results



Figure 5.12: SHAPENET DATASET. Nearest Neighbor Results

Chapter 6

Conclusions and Future Work

6.1 Conclusions

The central objective of this dissertation is to devise methodologies for generating high-resolution microstructure generation and synthetic 3D grain generation for the application of 3D microstructure generation. This endeavor aims to expedite the exploration of the process-structure-property (PSP) relationship, thereby facilitating the discovery of new materials while conserving time, labor, and capital. We first discuss microstructure data representation which are significantly different from common computer vision dataset. The microstructure dataset contains 3D crystal orientation information at each voxel which follows physics principles from crystallography. A Inverse Pole Figure (IPF) mapping technique different from RGB color coding is used to visualize microstructure dataset.

In this thesis, we discuss two key problems for the purpose of improving the exploration of the PSP relationship. The first involves the generation of high-resolution 3D microstructures using a physics-informed machine learning approach. A challenge here is the limited availability of experimental 3D microstructure datasets. To overcome this, we

have integrated prior physics knowledge into both the network architecture and the loss functions of machine learning algorithms. The crystal symmetry and quaternion orientation information are used in machine learning frameworks. Unlike existing SR methods which operate on scalar image data, the training pipeline is implemented in quaternion orientation space. The inference pipeline produces quaternion output that is converted into Euler angle representation and colored based on IPF projection conventions. We demonstrate that a network underpinned by physics principles outperforms traditional neural networks in synthesizing high-resolution microstructure datasets.

The second problem tackled in this dissertation concerns the generation of synthetic 3D grains for synthetic 3D microstructure generation. Creating individual grains with specific crystal orientations is less complex than synthesizing entire 3D microstructures, especially given the scarcity of comprehensive 3D microstructure datasets. To address this, we have developed the first known generative adversarial network (GAN) capable of realistically synthesizing individual grains from a latent space. This novel approach establishes a foundational framework for network-based methods in understanding the intricate relationships between grain structure and the properties of crystalline materials.

Through this research, we aim not only to advance the field of material science but also to provide a more efficient and resource-effective pathway for the exploration and development of new materials. Our work, at its core, seeks to bridge the gap between theoretical knowledge and practical application in the realm of microstructure analysis. We have provided our machine learning inference model on BisQue, an open source cloud infrastructure.

6.2 Future Work

For 3D microstructure generation, generating isolated synthetic grains is not sufficient. We require an advanced algorithm capable of constructing a continuous 3D synthetic microstructure from these grains, while adhering to the governing physical principles of grain boundary interactions, which combined with material processing steps, contribute to the grain morphology. Current methods [6, 172] often involve employing random space-filling methods with simplified grain shapes like spheres or ellipsoids, overlooking the intrinsic relationships between grains and actual shapes of grain boundaries.

One promising approach is to leverage large language models [173] to guide the assembly of synthetic grains using physical principles, such as minimizing boundary energy, ensuring a more precise representation. A pivotal element is leveraging latent space operations in Large Language Models (LLMs). Latent space refers to a condensed, feature-rich domain where complex data relationships are more tractable. The choice of latent space not only streamlines synthesis but also has the potential to elevate the quality of the resulting synthetic 3D microstructures. In the following, we will introduce examples of physics-based relationships that can be applied to generative models for synthetic microstructures, encompassing techniques such as GANs, variational autoencoders, diffusion models, and large language models. These approaches aim to further enhance the accuracy and efficiency of synthetic microstructure generation for materials science.

Our initial step as shown in figure involves creating individual synthetic 3D grains, each characterized by localized crystal orientation data. This task is simplified when approached through generative models rather than attempting to synthesize an entire 3D microstructure directly. Here we will leverage our prior work [94] on using GANs and explore the coupling with language models to cluster and describe the size and shape of such grains - like words in a sentence in natural language processing. There is no well

defined vocabulary that exists today, so we anticipate this to be an important first step in creating a language model for materials microstructures.

In the second step as shown in figure, we tackle the challenge of assembling these individual synthetic grains into a cohesive 3D microstructure. Conventional algorithms [6,172] often rely on random space-filling techniques using standard geometric shapes like ellipsoids and spheres to represent grains. Our approach involves developing a learning-based algorithm that leverages large language models (LLM) to synthesize these single grains. This approach will take into account the boundary relationships between adjacent grains. Given the high-dimensional nature of grain datasets, we plan to represent individual 3D grains within a latent space using an Encoder-Decoder architecture, enabling efficient encoding/decoding into 3D grains.

The third step focuses on identifying the optimal arrangement of neighboring grains for a single grain within the latent space, guided by processing parameters. LLMs will be trained to discern the relative ordering of grains that minimizes boundary energy [174], facilitating grain synthesis through a Decoder. Utilizing the ordered 3D grains, we will be equipped to synthesize the final 3D microstructure in the last step.

In the final Step 4, we will use the latent vectors that have embedded relative positional information, to initiate the generation of 3D grains utilizing our decoder. Subsequently, we will employ a space-filling algorithm to arrange these 3D grains within the prescribed volume. This process will result in the reconstruction of the 3D microstructure, incorporating the knowledge of learned grain and crystal orientation, ensuring their physical feasibility.

In summary, the four steps described above enable the creation of 3D synthetic microstructures that accurately embody both local crystal orientation and grain neighbor information. This achievement is made possible through the use of physics-guided generative models. By employing this method, we can generate 3D synthetic microstruc-

tures that not only exhibit desired physical properties but also mirror those that can be achieved through real-world material design processes.

The ability to generate such 3D synthetic microstructures significantly accelerates the exploration of process-structure-property (PSP) relationships. Leveraging generative models to access a diverse array of 3D synthetic microstructures under varying processing conditions alleviates the traditional constraints faced in materials discovery, including the slow pace of experimentation, as well as the high costs, labor, and capital requirements. This method offers a promising pathway to more efficient and innovative material development.

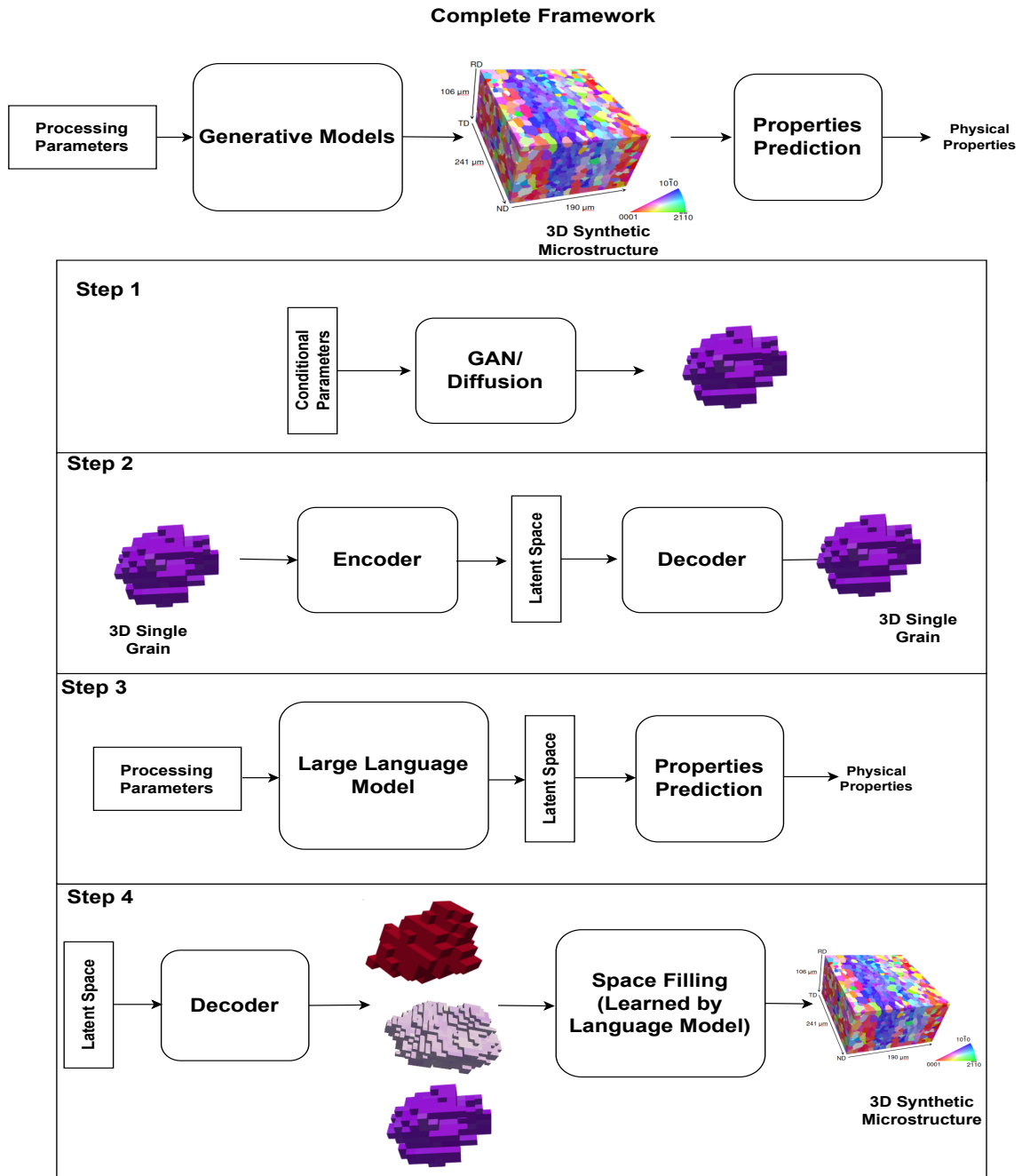


Figure 6.1: SYNTHETIC 3D MICROSTRUCTURE GENERATION PROCESS: In our framework, we want to generate 3D synthetic microstructure from given processing parameters which satisfies certain physical properties. In the first step, synthetic grains with crystal orientation information are generated using generative models from given parameters. In the second step, grains are encoded into latent space. In step 3, we learn optimal arrangement of grains using large language models in latent space. In step 4, latent spaces are decoded into grains and packed into a 3D volume to form 3D microstructure.

Bibliography

- [1] S. Hémerly, A. Naït-Ali, M. Guéguen, J. Wendorf, A. Polonsky, M. Echlin, J. Stinville, T. Pollock, and P. Villechaise, *A 3d analysis of the onset of slip activity in relation to the degree of micro-texture in ti-6al-4v*, *Acta Materialia* **181** (Dec., 2019) 36–48.
- [2] M. Charpagne, J. C. Stinville, A. T. Polonsky, M. P. Echlin, and T. M. Pollock, *A multi-modal data merging framework for correlative investigation of strain localization in three dimensions*, *JOM* **73** (Sept., 2021) 3263–3271.
- [3] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, *Generative adversarial nets*, in *Advances in Neural Information Processing Systems 27* (Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, eds.), pp. 2672–2680. Curran Associates, Inc., 2014.
- [4] J. Wu, C. Zhang, T. Xue, B. Freeman, and J. Tenenbaum, *Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling*, in *Advances in Neural Information Processing Systems 29* (D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, eds.), pp. 82–90. Curran Associates, Inc., 2016.
- [5] R. Quey, P. Dawson, and F. Barbe, *Large-scale 3d random polycrystals for the finite element method: Generation, meshing and remeshing*, *Computer Methods in Applied Mechanics and Engineering* **200** (2011), no. 17-20 1729–1745.
- [6] M. A. Groeber and M. A. Jackson, *Dream. 3d: a digital representation environment for the analysis of microstructure in 3d*, *Integrating materials and manufacturing innovation* **3** (2014), no. 1 56–72.
- [7] A. E. Robertson, C. Kelly, M. Buzzy, and S. R. Kalidindi, *Local-global decompositions for conditional microstructure generation*, *Acta Materialia* (2023) 118966.
- [8] D. K. Jangid, N. R. Brodrik, M. P. Echlin, C. Gudavalli, C. Levenson, T. M. Pollock, S. H. Daly, and B. Manjunath, *Q-rbsa: high-resolution 3d ebsd map*

- generation using an efficient quaternion transformer network, *npj Computational Materials* **10** (2024), no. 1 27.
- [9] A. Senthilnathan, P. Acar, and M. D. Graef, *Markov random field based microstructure reconstruction using the principal image moments*, *Materials Characterization* **178** (Aug., 2021) 111281.
- [10] T. Hsu, W. K. Epting, H. Kim, H. W. Abernathy, G. A. Hackett, A. D. Rollett, P. A. Salvador, and E. A. Holm, *Microstructure generation via generative adversarial network for heterogeneous, topologically complex 3d materials*, *JOM* **73** (Dec., 2020) 90–102.
- [11] A. Brust, E. Payton, T. Hobbs, V. Sinha, V. Yardley, and S. Niezgod, *Probabilistic reconstruction of austenite microstructure from electron backscatter diffraction observations of martensite*, *Microscopy and Microanalysis* **27** (Sept., 2021) 1035–1055.
- [12] H. Cao, C. Tan, Z. Gao, Y. Xu, G. Chen, P.-A. Heng, and S. Z. Li, *A survey on generative diffusion model*, *arXiv preprint arXiv:2209.02646* (2022).
- [13] J. Ho, A. Jain, and P. Abbeel, *Denoising diffusion probabilistic models*, *Advances in neural information processing systems* **33** (2020) 6840–6851.
- [14] K. Kvilekval, D. Fedorov, B. Obara, A. Singh, and B. Manjunath, *Bisque: a platform for bioimage analysis and management*, *Bioinformatics* **26** (2010), no. 4 544–552.
- [15] D. K. Jangid, N. R. Brodrik, M. P. Echlin, S. Daly, T. Pollock, and B. Manjunath, *Titanium 3d microstructure for physics-based generative models: a dataset and primer*, in *1st Workshop on the Synergy of Scientific and Machine Learning Modeling@ ICML2023*, 2023.
- [16] N. R. Council, *Integrated Computational Materials Engineering: A Transformational Discipline for Improved Competitiveness and National Security*. The National Academies Press, Washington, DC, 2008.
- [17] A. J. Schwartz, M. Kumar, B. L. Adams, and D. P. Field, *Electron backscatter diffraction in materials science*, *Electron Backscatter Diffraction in Materials Science* (2009) 1–403.
- [18] N. C. Krieger Lassen, D. Juul Jensen, and K. Conradsen, *Image processing procedures for analysis of electron back scattering patterns*, *Scanning microscopy* **6** (1992), no. 1 115–121.
- [19] B. L. Adams, S. I. Wright, and K. Kunze, *Orientation imaging: The emergence of a new microscopy*, *Metallurgical Transactions A* **24** (Apr., 1993) 819–831.

- [20] M. A. Jackson, E. Pascal, and M. D. Graef, *Dictionary indexing of electron back-scatter diffraction patterns: a hands-on tutorial*, *Integrating Materials and Manufacturing Innovation* **8** (May, 2019) 226–246.
- [21] W. Lenthe, S. Singh, and M. D. Graef, *A spherical harmonic transform approach to the indexing of electron back-scattered diffraction patterns*, *Ultramicroscopy* **207** (Dec., 2019) 112841.
- [22] Z. Ding, C. Zhu, and M. D. Graef, *Determining crystallographic orientation via hybrid convolutional neural network*, *Materials Characterization* **178** (Aug., 2021) 111213.
- [23] M. P. Echlin, A. T. Polonsky, J. Lamb, R. Geurts, S. J. Randolph, A. Botman, and T. M. Pollock, *Recent developments in femtosecond laser-enabled TriBeam systems*, *JOM* **73** (Nov., 2021) 4258–4269.
- [24] S. J. Randolph, J. Filevich, A. Botman, R. Gannon, C. Rue, and M. Straw, *In situ femtosecond pulse laser ablation for large volume 3d analysis in scanning electron microscope systems*, *Journal of Vacuum Science & Technology B* **36** (Nov., 2018) 06JB01.
- [25] D. Rowenhorst, A. D. Rollett, G. S. Rohrer, M. Groeber, M. Jackson, P. J. Konijnenberg, and M. D. Graef, *Consistent representations of and conversions between 3d rotations*, *Modelling and Simulation in Materials Science and Engineering* **23** (Oct., 2015) 083501.
- [26] M. De Graef, “Marc degraef’s 3d rotations github repository.” Accessed: 2023-05.
- [27] D. K. Jangid, N. R. Brodnik, M. G. Goebel, A. Khan, S. Majeti, M. P. Echlin, S. H. Daly, T. M. Pollock, and B. Manjunath, *Adaptable physics-based super-resolution for electron backscatter diffraction maps*, *npj Computational Materials* **8** (2022), no. 1 255.
- [28] T. Parcollet, Y. Zhang, M. Morchid, C. Trabelsi, G. Linares, R. De Mori, and Y. Bengio, *Quaternion convolutional neural networks for end-to-end automatic speech recognition*, *arXiv preprint arXiv:1806.07789* (2018).
- [29] D. Roşca, A. Morawiec, and M. D. Graef, *A new method of constructing a grid in the space of 3d rotations and its applications to texture analysis*, *Modelling and Simulation in Materials Science and Engineering* **22** (Oct., 2014) 075013.
- [30] A. Morawiec and D. Field, *Rodrigues parameterization for orientation and misorientation distributions*, *Philosophical Magazine A* **73** (1996), no. 4 1113–1130.

- [31] G. Nolze and R. Hielscher, *Orientations – perfectly colored*, *Journal of Applied Crystallography* **49** (Sept., 2016) 1786–1802.
- [32] M. P. Echlin, M. Straw, S. Randolph, J. Filevich, and T. M. Pollock, *The TriBeam system: Femtosecond laser ablation in situ SEM*, *Materials Characterization* **100** (Feb., 2015) 1–12.
- [33] M. P. Echlin, T. L. Burnett, A. T. Polonsky, T. M. Pollock, and P. J. Withers, *Serial sectioning in the SEM for three dimensional materials science*, *Current Opinion in Solid State and Materials Science* **24** (Apr., 2020) 100817.
- [34] S. Patala, J. K. Mason, and C. A. Schuh, *Improved representations of misorientation information for grain boundary science and engineering*, *Progress in Materials Science* **57** (2012) 1383–1425.
- [35] Z. Wang, J. Chen, and S. C. Hoi, *Deep learning for image super-resolution: A survey*, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **43** (10, 2021) 3365–3387.
- [36] J. Kim, J. K. Lee, and K. M. Lee, *Deeply-recursive convolutional network for image super-resolution*, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1637–1645, 2016.
- [37] Y. Tai, J. Yang, and X. Liu, *Image super-resolution via deep recursive residual network*, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3147–3155, 2017.
- [38] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee, *Enhanced deep residual networks for single image super-resolution*, in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 136–144, 2017.
- [39] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, *Image super-resolution using very deep residual channel attention networks*, in *Proceedings of the European conference on computer vision (ECCV)*, pp. 286–301, 2018.
- [40] T. Dai, J. Cai, Y. Zhang, S.-T. Xia, and L. Zhang, *Second-order attention network for single image super-resolution*, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11065–11074, 2019.
- [41] B. Niu, W. Wen, W. Ren, X. Zhang, L. Yang, S. Wang, K. Zhang, X. Cao, and H. Shen, *Single image super-resolution via a holistic attention network*, in *European Conference on Computer Vision*, pp. 191–207, Springer, 2020.
- [42] A. J. Schwartz, M. Kumar, B. L. Adams, and D. P. Field, eds., *Electron Backscatter Diffraction in Materials Science*. Springer US, Boston, MA, 2009.

- [43] K. Lassen, *Automatic high-precision measurements of the location and width of kikuchi bands in electron backscatter diffraction patterns*, *Journal of Microscopy* **190** (jun, 1998) 375–391.
- [44] Y. H. Chen, S. U. Park, D. Wei, G. Newstadt, M. A. Jackson, J. P. Simmons, M. De Graef, and A. O. Hero, *A dictionary approach to electron backscatter diffraction indexing*, *Microscopy and Microanalysis* **21** (jun, 2015) 739–752.
- [45] M. A. Jackson, E. Pascal, and M. De Graef, *Dictionary indexing of electron back-scatter diffraction patterns: a hands-on tutorial*, *Integrating Materials and Manufacturing Innovation* **8** (may, 2019) 226–246.
- [46] W. J. Sames, F. A. List, S. Pannala, R. R. Dehoff, and S. S. Babu, *The metallurgy and processing science of metal additive manufacturing*, *International Materials Reviews* **61** (Mar., 2016) 315–360.
- [47] L. N. Brewer, D. P. Field, and C. C. Merriman, *Mapping and assessing plastic deformation using EBSD*, in *Electron Backscatter Diffraction in Materials Science*, pp. 251–262. Springer US, Boston, MA, 2009.
- [48] W. A. Witzel, A. T. Polonsky, T. M. Pollock, and I. J. Beyerlein, *Three-dimensional maps of geometrically necessary dislocation densities in additively manufactured Ni-based superalloy IN718*, *International Journal of Plasticity* **131** (Aug., 2020) 102709.
- [49] W. A. Witzel, A. T. Polonsky, P. F. Rottmann, K. M. Pusch, M. P. Echlin, T. M. Pollock, and I. J. Beyerlein, *Boundary characterization using 3d mapping of geometrically necessary dislocations in AM Ta microstructure*, *Journal of Materials Science* (Mar., 2022).
- [50] S. Ren, E. Kenik, K. Alexander, and A. Goyal, *Exploring Spatial Resolution in Electron Back-Scattered Diffraction Experiments via Monte Carlo Simulation*, *Microscopy and Microanalysis* **4** (1998), no. 1 15–22.
- [51] D. Chen, J. C. Kuo, and W. T. Wu, *Effect of microscopic parameters on EBSD spatial resolution*, *Ultramicroscopy* **111** (aug, 2011) 1488–1494.
- [52] R. Keller and R. Geiss, *Transmission EBSD from 10 nm domains in a scanning electron microscope*, *Journal of Microscopy* **245** (mar, 2012) 245–251.
- [53] D. R. Steinmetz and S. Zaefferer, *Towards ultrahigh resolution EBSD by low accelerating voltage*, *Materials Science and Technology* **26** (June, 2010) 640–645.
- [54] A. Tripathi and S. Zaefferer, *On the resolution of EBSD across atomic density and accelerating voltage with a particular focus on the light metal magnesium*, *Ultramicroscopy* **207** (dec, 2019) 112828.

- [55] K. Kaufmann, H. Lane, X. Liu, and K. S. Vecchio, *Efficient few-shot machine learning for classification of EBSD patterns*, *Scientific Reports* **11** (Apr., 2021).
- [56] K. Kaufmann, C. Zhu, A. S. Rosengarten, D. Maryanovsky, T. J. Harrington, E. Marin, and K. S. Vecchio, *Crystal symmetry determination in electron diffraction using machine learning*, *Science* **367** (Jan., 2020) 564–568.
- [57] J. Jung, J. Na, H. K. Park, J. M. Park, G. Kim, S. Lee, and H. S. Kim, *Super-resolving material microstructure image via deep learning for microstructure characterization and mechanical behavior analysis*, *npj Computational Materials* *2021 7:1* **7** (jun, 2021) 1–11.
- [58] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, *Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network*, 2016.
- [59] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, *Loss functions for neural networks for image processing*, 2018.
- [60] V. I. Skrytnyy and V. N. Yaltsev, *Relative misorientations of crystals*, *IOP Conference Series: Materials Science and Engineering* **130** (apr, 2016) 012059.
- [61] M. P. Echlin, T. L. Burnett, A. T. Polonsky, T. M. Pollock, and P. J. Withers, *Serial sectioning in the SEM for three dimensional materials science*, *Current Opinion in Solid State and Materials Science* **24** (Apr., 2020) 100817.
- [62] M. P. Miller, D. C. Pagan, A. J. Beaudoin, K. E. Nygren, and D. J. Shadle, *Understanding Micromechanical Material Behavior Using Synchrotron X-rays and In Situ Loading*, *Metallurgical and Materials Transactions A* *2020 51:9* **51** (jun, 2020) 4360–4376.
- [63] J. V. Bernier, R. M. Suter, A. D. Rollett, and J. D. Almer, *High-energy x-ray diffraction microscopy in materials science*, *Annual Review of Materials Research* **50** (July, 2020) 395–436.
- [64] P. Reischig and W. Ludwig, *Three-dimensional reconstruction of intragranular strain and orientation in polycrystals by near-field x-ray diffraction*, *Current Opinion in Solid State and Materials Science* **24** (2020), no. 5 100851.
- [65] D. J. Rowenhorst, L. Nguyen, A. D. Murphy-Leonard, and R. W. Fonda, *Characterization of microstructure in additively manufactured 316l using automated serial sectioning*, *Current Opinion in Solid State and Materials Science* **24** (June, 2020) 100819.

- [66] M. G. Chapman, M. N. Shah, S. P. Donegan, J. M. Scott, P. A. Shade, D. Menasche, and M. D. Uchic, *AFRL Additive Manufacturing Modeling Series: Challenge 4, 3D Reconstruction of an IN625 High-Energy Diffraction Microscopy Sample Using Multi-modal Serial Sectioning, Integrating Materials and Manufacturing Innovation 2021 10:2* **10** (jun, 2021) 129–141.
- [67] A. Garner, J. Donoghue, R. Geurts, Y. A. Aboura, B. Winiarski, P. B. Prangnell, and T. L. Burnett, *Large-scale serial sectioning of environmentally assisted cracks in xxx al alloys using femtosecond laser-PFIB*, *Materials Characterization* **188** (June, 2022) 111890.
- [68] G. M. D. P. Godaliyadda, D. H. Ye, M. D. Uchic, M. A. Groeber, G. T. Buzzard, and C. A. Bouman, *A framework for dynamic image sampling based on supervised learning*, *IEEE Transactions on Computational Imaging* **4** (Mar., 2018) 1–16.
- [69] Y. Zhang, G. M. D. Godaliyadda, N. Ferrier, E. B. Gulsoy, C. A. Bouman, and C. Phatak, *Reduced electron exposure for energy-dispersive spectroscopy using dynamic sampling*, *Ultramicroscopy* **184** (Jan., 2018) 90–97.
- [70] V. S. Tong, A. J. Knowles, D. Dye, and T. B. Britton, *Rapid electron backscatter diffraction mapping: Painting by numbers*, *Materials Characterization* **147** (Jan., 2019) 271–279.
- [71] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, and M.-H. Yang, *Restormer: Efficient transformer for high-resolution image restoration*, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5728–5739, 2022.
- [72] D. K. Jangid, N. R. Brodник, M. G. Goebel, A. Khan, S. Majeti, M. P. Echlin, S. H. Daly, T. M. Pollock, and B. Manjunath, *Adaptable physics-based super-resolution for electron backscatter diffraction maps*, *npj Computational Materials* **8** (2022), no. 1 255.
- [73] C. Trabelsi, O. Bilaniuk, Y. Zhang, D. Serdyuk, S. Subramanian, J. Santos, S. Mehri, N. Rostamzadeh, Y. Bengio, and C. Pal, *Deep complex networks*. *arxiv preprint arxiv: 170509792*, .
- [74] I. Aizenberg and A. Gonzalez, *Image recognition using mlmvn and frequency domain features*, in *2018 International joint conference on neural networks (IJCNN)*, pp. 1–8, IEEE, 2018.
- [75] N. Matsui, T. Isokawa, H. Kusamichi, F. Peper, and H. Nishimura, *Quaternion neural network with geometrical operators*, *Journal of Intelligent & Fuzzy Systems* **15** (2004), no. 3-4 149–164.

- [76] H. Kusamichi, T. Isokawa, N. Matsui, Y. Ogawa, and K. Maeda, *A new scheme for color night vision by quaternion neural network*, in *Proceedings of the 2nd international conference on autonomous robots and agents*, vol. 1315, 2004.
- [77] T. Isokawa, N. Matsui, and H. Nishimura, *Quaternionic neural networks: Fundamental properties and applications*, in *Complex-valued neural networks: utilizing high-dimensional parameters*, pp. 411–439. IGI global, 2009.
- [78] X. Yun and E. R. Bachmann, *Design, implementation, and experimental results of a quaternion-based kalman filter for human body motion tracking*, *IEEE transactions on Robotics* **22** (2006), no. 6 1216–1227.
- [79] K. Shoemake, *Animating rotation with quaternion curves*, in *Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, pp. 245–254, 1985.
- [80] D. Pletinckx, *Quaternion calculus as a basic tool in computer graphics*, *The Visual Computer* **5** (1989), no. 1 2–13.
- [81] X. Zhu, Y. Xu, H. Xu, and C. Chen, *Quaternion convolutional neural networks*, in *Proceedings of the European Conference on Computer Vision (ECCV)*, September, 2018.
- [82] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, *Attention is all you need*, *Advances in neural information processing systems* **30** (2017).
- [83] J. Liang, J. Cao, G. Sun, K. Zhang, L. Van Gool, and R. Timofte, *Swinir: Image restoration using swin transformer*, in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1833–1844, 2021.
- [84] T. Parcollet, M. Morchid, and G. Linarès, *A survey of quaternion neural networks*, *Artificial Intelligence Review* **53** (2020), no. 4 2957–2982.
- [85] C. J. Gaudet and A. S. Maida, *Deep quaternion networks*, in *2018 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, IEEE, 2018.
- [86] A. Krizhevsky, I. Sutskever, and G. E. Hinton, *Imagenet classification with deep convolutional neural networks*, *Communications of the ACM* **60** (2017), no. 6 84–90.
- [87] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et. al.*, *An image is worth 16x16 words: Transformers for image recognition at scale*, *arXiv preprint arXiv:2010.11929* (2020).

- [88] D. Hendrycks and K. Gimpel, *Gaussian error linear units (gelus)*, *arXiv preprint arXiv:1606.08415* (2016).
- [89] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, *Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network*, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1874–1883, 2016.
- [90] S. De Leo and P. Rotelli, *Local hypercomplex analyticity*, *arXiv preprint funct-an/9703002* (1997).
- [91] T. Isokawa, H. Nishimura, and N. Matsui, *Quaternionic multilayer perceptron with local analyticity*, *Information* **3** (2012), no. 4 756–770.
- [92] B. C. Ujang, C. Jahanchahi, C. C. Took, and D. Mandic, *Quaternion valued neural networks and nonlinear adaptive filters*, *IEEE Trans. Neural Netw* (2010).
- [93] P. Arena, L. Fortuna, L. Occhipinti, and M. G. Xibilia, *Neural networks for quaternion-valued function approximation*, in *Proceedings of IEEE International Symposium on Circuits and Systems-ISCAS'94*, vol. 6, pp. 307–310, IEEE, 1994.
- [94] D. K. Jangid, N. R. Brodnik, A. Khan, M. G. Goebel, M. P. Echlin, T. M. Pollock, S. H. Daly, and B. Manjunath, *3d grain shape generation in polycrystals using generative adversarial networks*, *Integrating Materials and Manufacturing Innovation* **11** (2022) 71–84.
- [95] J. Oddershede, J. Sun, N. Gueninchault, F. Bachmann, H. Bale, C. Holzner, and E. Lauridsen, *Non-destructive characterization of polycrystalline materials in 3d by laboratory diffraction contrast tomography*, *Integrating Materials and Manufacturing Innovation* **8** (May, 2019) 217–225.
- [96] F. Bachmann, H. Bale, N. Gueninchault, C. Holzner, and E. M. Lauridsen, *3d grain reconstruction from laboratory diffraction contrast tomography*, *Journal of Applied Crystallography* **52** (May, 2019) 643–651.
- [97] National Research Council, *Integrated Computational Materials Engineering: A Transformational Discipline for Improved Competitiveness and National Security*. The National Academies Press, Washington, DC, 2008.
- [98] National Science and Technology Council (US), *Materials genome initiative for global competitiveness*. Executive Office of the President, National Science and Technology Council, 2011.
- [99] X.-G. Lu, *Remarks on the recent progress of Materials Genome Initiative*, *Science Bulletin* 2015 60:22 **60** (nov, 2015) 1966–1968.

- [100] M. I. I. M2I, *ICMEG: Integrated Computational Materials Engineering Expert Group*, 2016.
- [101] A. Jain, K. A. Persson, and G. Ceder, *Research Update: The materials genome initiative: Data sharing and the impact of collaborative ab initio databases*, *APL Materials* **4** (mar, 2016) 053102.
- [102] C. Draxl and M. Scheffler, *NOMAD: The FAIR Concept for Big-Data-Driven Materials Science*, *MRS Bulletin* **43** (may, 2018) 676–682, [arXiv:1805.0503].
- [103] J. Pennekamp, R. Glebke, M. Henze, T. Meisen, C. Quix, R. Hai, L. Gleim, P. Niemietz, M. Rudack, S. Knape, A. Epple, D. Trauth, U. Vroomen, T. Bergs, C. Brecher, A. Buhrig-Polaczek, M. Jarke, and K. Wehrle, *Towards an infrastructure enabling the internet of production*, *Proceedings - 2019 IEEE International Conference on Industrial Cyber Physical Systems, ICPS 2019* (may, 2019) 31–37.
- [104] D. J. Jensen, E. M. Lauridsen, L. Margulies, H. F. Poulsen, S. Schmidt, H. O. Sørensen, and G. B. Vaughan, *X-ray microscopy in four dimensions*, *Materials Today* **9** (jan, 2006) 18–25.
- [105] W. Ludwig, P. Reischig, A. King, M. Herbig, E. M. Lauridsen, G. Johnson, T. J. Marrow, and J. Y. Buffière, *Three-dimensional grain mapping by x-ray diffraction contrast tomography and the use of Friedel pairs in diffraction data analysis*, *Review of Scientific Instruments* **80** (mar, 2009) 033905.
- [106] H. Poulsen, *An introduction to three-dimensional X-ray diffraction microscopy*, *urn:issn:0021-8898* **45** (oct, 2012) 1084–1097.
- [107] M. P. Echlin, A. Mottura, C. J. Torbet, and T. M. Pollock, *A new TriBeam system for three-dimensional multimodal materials analysis*, *Review of Scientific Instruments* **83** (feb, 2012) 023701.
- [108] S. Hata, T. Honda, H. Saito, M. Mitsuhara, T. C. Petersen, and M. Murayama, *Electron tomography: An imaging method for materials deformation dynamics*, *Current Opinion in Solid State and Materials Science* **24** (aug, 2020) 100850.
- [109] D. J. Rowenhorst, L. Nguyen, A. D. Murphy-Leonard, and R. W. Fonda, *Characterization of microstructure in additively manufactured 316l using automated serial sectioning*, *Current Opinion in Solid State and Materials Science* **24** (June, 2020) 100819.
- [110] S. Chaudhuri, E. Kalogerakis, L. Guibas, and V. Koltun, *Probabilistic reasoning for assembly-based 3D modeling*, *ACM Transactions on Graphics* **30** (jul, 2011) 1.

- [111] T. Funkhouser, M. Kazhdan, P. Shilane, P. Min, W. Kiefer, A. Tal, S. Rusinkiewicz, and D. Dobkin, *Modeling by example*, *ACM Transactions on Graphics* **23** (Aug., 2004) 652–663.
- [112] E. Kalogerakis, S. Chaudhuri, D. Koller, and V. Koltun, *A probabilistic model for component-based shape synthesis*, *ACM Transactions on Graphics (TOG)* **31** (2012), no. 4 1–11.
- [113] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, *3d shapenets: A deep representation for volumetric shapes*, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June, 2015.
- [114] C. B. Choy, D. Xu, J. Gwak, K. Chen, and S. Savarese, *3d-r2n2: A unified approach for single and multi-view 3d object reconstruction*, in *European conference on computer vision*, pp. 628–644, Springer, 2016.
- [115] X. Zhang, Z. Zhang, C. Zhang, J. B. Tenenbaum, W. T. Freeman, and J. Wu, *Learning to reconstruct shapes from unseen classes*, in *Proceedings of the 32nd International Conference on Neural Information Processing Systems, NIPS’18*, (Red Hook, NY, USA), p. 2263–2274, Curran Associates Inc., 2018.
- [116] J. Wu, Y. Wang, T. Xue, X. Sun, B. Freeman, and J. Tenenbaum, *Marrnet: 3d shape reconstruction via 2.5d sketches*, in *Advances in Neural Information Processing Systems 30* (I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds.), pp. 540–550. Curran Associates, Inc., 2017.
- [117] A. Noguchi and T. Harada, *Rgbd-gan: Unsupervised 3d representation learning from natural image datasets via rgbd image synthesis*, *arXiv preprint arXiv:1909.12573* (2019).
- [118] T. Nguyen-Phuoc, C. Li, L. Theis, C. Richardt, and Y.-L. Yang, *Hologan: Unsupervised learning of 3d representations from natural images*, in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 7588–7597, 2019.
- [119] A. Sharma, O. Grau, and M. Fritz, *Vconv-dae: Deep volumetric shape learning without object labels*, in *European Conference on Computer Vision*, pp. 236–250, Springer, 2016.
- [120] J. Zhu, J. Xie, and Y. Fang, *Learning adversarial 3d model generation with 2d image enhancer*, .
- [121] M. Coster, X. Arnould, J.-L. Chermant, A. E. Moataz, and T. Chartier, *A Microstructural Model by Space Tessellation for a Sintered Ceramic: Cerine*, *Image Analysis & Stereology* **24** (may, 2005) 105.

- [122] W. Johnson and R. Mehl, *Reaction kinetics in processes of nucleation and growth*, *Transactions of the Metallurgical Society of AIME* **135** (1939) 416–442.
- [123] M. Avrami, *Kinetics of phase change. I general theory*, *The Journal of Chemical Physics* **7** (1939), no. 12 1103–1112.
- [124] A. N. Kolmogorov, *A statistical theory for the recrystallization of metals*, *Izvestiya Rossiiskoi Akademii Nauk. Seriya Matematicheskaya* (1937).
- [125] M. Nosonovsky, X. Zhang, and S. K. Esche, *Related content Scaling of Monte Carlo simulations of grain growth in metals*, *Modelling and Simulation in Materials Science and Engineering* **17** (2009) 1–13.
- [126] M. A. Groeber and M. A. Jackson, *DREAM.3D: A Digital Representation Environment for the Analysis of Microstructure in 3D*, *Integrating Materials and Manufacturing Innovation* **3** (dec, 2014) 56–72.
- [127] P. G. Callahan, M. Groeber, and M. De Graef, *Towards a quantitative comparison between experimental and synthetic grain structures*, *Acta Materialia* **111** (jun, 2016) 242–252.
- [128] V. Bapst, T. Keck, A. Grabska-Barwińska, C. Donner, E. Cubuk, S. Schoenholz, A. Obika, A. Nelson, T. Back, D. Hassabis, and P. Kohli, *Unveiling the predictive power of static structure in glassy systems*, *Nature Physics* **16** (04, 2020) 448–454.
- [129] A. Cecen, H. Dai, Y. C. Yabansu, S. R. Kalidindi, and L. Song, *Material structure-property linkages using three-dimensional convolutional neural networks*, *Acta Materialia* **146** (mar, 2018) 76–84.
- [130] R. Bostanabad, A. T. Bui, W. Xie, D. W. Apley, and W. Chen, *Stochastic microstructure characterization and reconstruction via supervised learning*, *Acta Materialia* **103** (jan, 2016) 89–102.
- [131] X. Li, Z. Yang, L. C. Brinson, A. Choudhary, A. Agrawal, and W. Chen, *A deep adversarial learning methodology for designing microstructural material systems*, in *ASME 2018 international design engineering technical conferences and computers and information in engineering*, 2018.
- [132] B. L. DeCost, M. D. Hecht, T. Francis, B. A. Webler, Y. N. Picard, and E. A. Holm, *UHCSDB: UltraHigh Carbon Steel Micrograph DataBase*, *Integrating Materials and Manufacturing Innovation* **6** (jun, 2017) 197–205.
- [133] B. L. DeCost, T. Francis, and E. A. Holm, *Exploring the microstructure manifold: Image texture representations applied to ultrahigh carbon steel microstructures*, *Acta Materialia* **133** (jul, 2017) 30–40, [arXiv:1702.01111].

- [134] A. Iyer, B. Dey, A. Dasgupta, W. Chen, and A. Chakraborty, *A conditional generative model for predicting material microstructures from processing methods*, *arXiv preprint arXiv:1910.02133* (2019).
- [135] D. Fokina, E. Muravleva, G. Ovchinnikov, and I. Oseledets, *Microstructure synthesis using style-based generative adversarial networks*, *Physical Review E* **101** (apr, 2020) 043308, [arXiv:1909.0704].
- [136] R. Cang, Y. Xu, S. Chen, Y. Liu, Y. Jiao, and M. Y. Ren, *Microstructure Representation and Reconstruction of Heterogeneous Materials Via Deep Belief Network for Computational Material Design*, *Journal of Mechanical Design, Transactions of the ASME* **139** (jul, 2017) [arXiv:1612.0740].
- [137] S. Torquato, *Random Heterogeneous Materials*, vol. 16 of *Interdisciplinary Applied Mathematics*. Springer New York, New York, NY, 2002.
- [138] R. Liu, A. Kumar, Z. Chen, A. Agrawal, V. Sundararaghavan, and A. Choudhary, *A predictive machine learning approach for microstructure optimization and materials design*, *Nature Scientific Reports* **5** (jun, 2015) 1–12.
- [139] T. Hsu, W. K. Epting, H. Kim, H. W. Abernathy, G. A. Hackett, A. D. Rollett, P. A. Salvador, and E. A. Holm, *Microstructure Generation via Generative Adversarial Network for Heterogeneous, Topologically Complex 3D Materials*, *JOM* **73** (jan, 2021) 90–102, [arXiv:2006.1388].
- [140] W. H. Bragg, W. L. Bragg Apr, B. W. H Bragg, and C. Professor of Physics, *The reflection of X-rays by crystals*, *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character* **88** (jul, 1913) 428–438.
- [141] F. Humphreys, *Quantitative metallography by electron backscattered diffraction*, *Journal of microscopy* **195** (1999), no. 3 170–185.
- [142] A. J. Wilkinson and T. B. Britton, *Strains, planes, and EBSD in materials science*, *Materials Today* **15** (sep, 2012) 366–376.
- [143] Y. H. Chen, S. U. Park, D. Wei, G. Newstadt, M. A. Jackson, J. P. Simmons, M. De Graef, and A. O. Hero, *A Dictionary Approach to Electron Backscatter Diffraction Indexing*, *Microscopy and Microanalysis* **21** (apr, 2015) 739–752.
- [144] Z. Ding, E. Pascal, and M. De Graef, *Indexing of electron back-scatter diffraction patterns using a convolutional neural network*, *Acta Materialia* **199** (oct, 2020) 370–382.
- [145] T. Karras, S. Laine, and T. Aila, *A style-based generator architecture for generative adversarial networks*, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4401–4410, 2019.

- [146] Z. Lin, A. Khetan, G. Fanti, and S. Oh, *Pacgan: The power of two samples in generative adversarial networks*, in *Advances in neural information processing systems*, pp. 1498–1507, 2018.
- [147] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, *et. al.*, *Shapenet: An information-rich 3d model repository*, *arXiv preprint arXiv:1512.03012* (2015).
- [148] M. P. Echlin, M. Straw, S. Randolph, J. Filevich, and T. M. Pollock, *The TriBeam system: Femtosecond laser ablation in situ SEM*, *Materials Characterization* **100** (Feb., 2015) 1–12.
- [149] M. P. Echlin, J. C. Stinville, V. M. Miller, W. C. Lenthe, and T. M. Pollock, *Incipient slip and long range plastic strain localization in microtextured Ti-6Al-4V titanium*, *Acta Materialia* **114** (aug, 2016) 164–175.
- [150] K. Chatterjee, M. P. Echlin, M. Kasemer, P. G. Callahan, T. M. Pollock, and P. Dawson, *Prediction of tensile stiffness and strength of Ti-6Al-4V using instantiated volume elements and crystal plasticity*, *Acta Materialia* **157** (sep, 2018) 21–32.
- [151] D. P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, *arXiv preprint arXiv:1412.6980* (2014).
- [152] M. Arjovsky, S. Chintala, and L. Bottou, *Wasserstein gan*, *arXiv preprint arXiv:1701.07875* (2017).
- [153] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, *Improved training of wasserstein gans*, *CoRR abs/1704.00028* (2017) [arXiv:1704.0002].
- [154] J. P. MacSleyne, J. P. Simmons, and M. D. Graef, *On the use of moment invariants for the automated analysis of 3D particle shapes*, *Modelling and Simulation in Materials Science and Engineering* **16** (apr, 2008) 045008.
- [155] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz, *Rotation invariant spherical harmonic representation of 3 d shape descriptors*, in *Symposium on geometry processing*, vol. 6, pp. 156–164, 2003.
- [156] A. Trenkle, M. Syha, W. Rheinheimer, P. G. Callahan, L. Nguyen, W. Ludwig, W. Lenthe, M. P. Echlin, T. M. Pollock, D. Weygand, M. Graef, M. J. Hoffmann, and P. Gumbsch, *Nondestructive evaluation of 3D microstructure evolution in strontium titanate*, *Journal of Applied Crystallography* **53** (apr, 2020) 349–359.
- [157] D. Maturana and S. Scherer, *Voxnet: A 3d convolutional neural network for real-time object recognition*, in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 922–928, Sep., 2015.

- [158] A. Sinha, J. Bai, and K. Ramani, *Deep learning 3d shape surfaces using geometry images*, in *ECCV*, 2016.
- [159] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, *Pointnet: Deep learning on point sets for 3d classification and segmentation*, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 652–660, 2017.
- [160] S. Bai, X. Bai, Z. Zhou, Z. Zhang, and L. Jan Latecki, *Gift: A real-time and scalable 3d shape search engine*, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5023–5032, 2016.
- [161] V. Hegde and R. Zadeh, *Fusionnet: 3d object classification using multiple data representations*, *arXiv preprint arXiv:1607.05695* (2016).
- [162] L. Kobbelt, P. Schröder, M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz, *Rotation invariant spherical harmonic representation of 3d shape descriptors*, *Proc 2003 Eurographics* vol. **43** (07, 2003).
- [163] D.-Y. Chen, X.-P. Tian, Y.-T. Shen, and M. Ouhyoung, *On visual similarity based 3d model retrieval*, in *Computer graphics forum*, vol. 22, pp. 223–232, Wiley Online Library, 2003.
- [164] Z. Wang, J. Chen, and S. C. Hoi, *Deep learning for image super-resolution: A survey*, *IEEE transactions on pattern analysis and machine intelligence* **43** (2020), no. 10 3365–3387.
- [165] C. H. Lin, H.-Y. Lee, Y.-C. Cheng, S. Tulyakov, and M.-H. Yang, *InfinityGAN: Towards Infinite-Resolution Image Synthesis*, arXiv:2104.0396.
- [166] N. Gey, P. Bocher, E. Uta, L. Germain, and M. Humbert, *Texture and microtexture variations in a near- α titanium forged disk of bimodal microstructure*, *Acta Materialia* **60** (apr, 2012) 2647–2655.
- [167] A. L. Pilchak, C. J. Szczepanski, J. A. Shaffer, A. A. Salem, and S. L. Semiatin, *Characterization of Microstructure, Texture, and Microtexture in Near-Alpha Titanium Mill Products*, *Metallurgical and Materials Transactions A* 2013 *44:11* **44** (jun, 2013) 4881–4890.
- [168] S. L. Semiatin, *An Overview of the Thermomechanical Processing of α/β Titanium Alloys: Current Status and Future Research Opportunities*, *Metallurgical and Materials Transactions A* 2020 *51:6* **51** (jan, 2020) 2593–2625.
- [169] R. R. Dehoff, M. M. Kirka, F. A. List, K. A. Unocic, and W. J. Sames, *Crystallographic texture engineering through novel melt strategies via electron beam melting: Inconel 718*, <https://doi.org/10.1179/1743284714Y.0000000697> **31** (jun, 2014) 939–944.

- [170] A. T. Polonsky, N. Raghavan, M. P. Echlin, M. M. Kirka, R. R. Dehoff, and T. M. Pollock, *3D Characterization of the Columnar-to-Equiaxed Transition in Additively Manufactured Inconel 718*, *Minerals, Metals and Materials Series* (2020) 990–1002.
- [171] S. P. Murray, K. M. Pusch, A. T. Polonsky, C. J. Torbet, G. G. E. Seward, N. Zhou, S. A. J. Forsik, P. Nandwana, M. M. Kirka, R. R. Dehoff, W. E. Slye, and T. M. Pollock, *A defect-resistant Co–Ni superalloy for 3D printing*, *Nature Communications 2020 11:1* **11** (oct, 2020) 1–11.
- [172] R. Quey and M. Kasemer, *The Neper/FEPX Project: Free / Open-source Polycrystal Generation, Deformation Simulation, and Post-processing*, *IOP Conference Series: Materials Science and Engineering* **1249** (July, 2022) 012021.
- [173] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong, *et. al.*, *A survey of large language models*, *arXiv preprint arXiv:2303.18223* (2023).
- [174] D. M. Saylor and G. S. Rohrer, *Evaluating anisotropic surface energies using the capillarity vector reconstruction method*, *Interface Science* **9** (2001) 35–42.