



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΜΕ
ΕΦΑΡΜΟΓΕΣ ΣΤΗ ΒΙΟΪΑΤΡΙΚΗ

Industrial Process Optimisation using Artificial Intelligence

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of
Philosophy

Konstantina Malamousi

January 2024



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΜΕ
ΕΦΑΡΜΟΓΕΣ ΣΤΗ ΒΙΟΪΑΤΡΙΚΗ

**Βελτιστοποίηση Βιομηχανικών Διεργασιών Με Χρήση Τεχνητής
Νοημοσύνης**

Διατριβή η οποία υποβλήθηκε για την εκπλήρωση των υποχρεώσεων απόκτησης του
Διδακτορικού Διπλώματος

Κωνσταντίνα Μαλαμούση

Επιβλέπων: Κωνσταντίνος Δελέμπασης, Καθηγητής

Ιανουάριος 2024



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΜΕ
ΕΦΑΡΜΟΓΕΣ ΣΤΗ ΒΙΟΪΑΤΡΙΚΗ

Industrial Process Optimisation using Artificial Intelligence

PhD Dissertation

Malamousi Konstantina

Advisory committee

Konstantinos Delibasis, Professor, Univ. of Thessaly (Supervisor)

Vassilios Plagianakos, Professor, Univ. of Thessaly

Sotirios Tasoulis, Assistant Professor, Univ. of Thessaly

Examination committee

Konstantinos Delibasis, Professor, Univ. of Thessaly (Supervisor)

Vassilios Plagianakos, Professor, Univ. of Thessaly

Sotirios Tasoulis, Associate Professor, Univ. of Thessaly

Alexandros Karantzalis, Professor, Univ. of Ioannina

Theodoros Karakasidis, Professor, Univ. of Thessaly

Maria Adam, Professor, Univ. of Thessaly

Panagiotis Vartholomeos, Assisatnt Professor, Univ. of Thessaly

DISCLAIMER ON ACADEMIC ETHICS AND INTELLECTUAL PROPERTY RIGHTS

Being fully aware of the implications of copyright laws, I expressly state that this PH.D. dissertation, as well as the electronic files and source codes developed or modified in the course of this thesis, are solely the product of my personal work and do not infringe any rights of intellectual property, personality and personal data of third parties, do not contain work / contributions of third parties for which the permission of the authors / beneficiaries is required and are not a product of partial or complete plagiarism, while the sources used are limited to the bibliographic references only and meet the rules of scientific citing. The points where I have used ideas, text, files and / or sources of other authors are clearly mentioned in the text with the appropriate citation and the relevant complete reference is included in the bibliographic references section. I also declare that the results of the work have not been used to obtain another degree. I fully, individually and personally undertake all legal and administrative consequences that may arise in the event that it is proven, in the course of time, that this thesis or part of it does not belong to me because it is a product of plagiarism.

The declarant

Konstantina Malamousi

Acknowledgement

I would like to take this opportunity to express my deepest gratitude to my supervisor, Prof. Konstantinos Delibasis, for his invaluable guidance, encouragement, and support. His provision of essential resources was instrumental in the completion of this work. I also extend my sincere thanks to the advisory committee for their support throughout this project. Special appreciation goes to my husband for his constant support and encouragement, as well as to my family as a whole. Finally, I would like to dedicate this thesis to my 8-year-old son, Christoforos, in recognition of the patience he has shown during the course of this project.

Abstract

The integration of Artificial Intelligence (AI) in industrial processes is a transformative development, reshaping the landscape of modern manufacturing and production. This thesis, titled (Optimization of Industrial Processes Using Artificial Intelligence), seeks to demonstrate the vast potential of AI in enhancing the efficiency, quality, and adaptability of various industrial processes. It begins by establishing a foundation in the broad application of AI across diverse sectors, highlighting how these technologies are revolutionizing traditional practices through improved decision-making, predictive analytics, and process optimization. The thesis then narrows its focus to a specific industrial process – thermal and cold spray technology in surface engineering. This sector, significant for its role in product enhancement and functionality, is on the cusp of a major shift thanks to the digital transformation driven by AI and Industrial Digital Technologies (IDTs). The study reviews existing digital tools, identifies key challenges, and proposes groundbreaking machine learning-based solutions for optimizing coating deposition, process control, and equipment health monitoring. A significant portion of the thesis is devoted to the implementation of advanced methodologies like convolutional neural networks for meticulous process optimization and monitoring. These methodologies represent a major departure from traditional methods, offering enhanced precision and adaptability. Furthermore, the thesis explores the relatively untapped potential of applying computer vision in spray monitoring. This innovative approach promises to revolutionize process analysis and quality control in thermal and cold spray technologies, highlighting the unexplored applications of AI in industrial processes. In conclusion, the thesis validates these novel AI approaches and provides implementation codes, effectively bridging the gap between theoretical innovation and practical application. The results demonstrate the transformative potential of AI in revolutionizing the flexibility, quality, and efficiency of not only thermal and cold spray processes but also broader industrial operations. This comprehensive study, therefore, contributes significantly to the field of industrial process optimization using artificial intelligence, setting new standards and opening new avenues for research and application. All experimental data were provided by Castolin Eutectic - Monitor Coatings Ltd.

Keywords: Artificial Intelligence, Machine Learning, CNNs, Thermal Spray, IoT

Περίληψη

Η ενσωμάτωση της Τεχνητής Νοημοσύνης (TN) στις βιομηχανικές διεργασίες αποτελεί μια εξέλιξη, η οποία αναδιαμορφώνει το τοπίο της σύγχρονης βιομηχανικής παραγωγής. Η παρούσα διατριβή, με τίτλο (Βελτιστοποίηση Βιομηχανικών Διαδικασιών μέσω της Τεχνητής Νοημοσύνης), επιχειρεί να αναδείξει το τεράστιο δυναμικό της TN στην ενίσχυση της αποδοτικότητας, ποιότητας και προσαρμοστικότητας διαφόρων βιομηχανικών διεργασιών. Ξεκινά με τη διαμόρφωση μιας βιβλιογραφικής βάσης στην ευρεία εφαρμογή της TN σε διάφορους τομείς, επισημαίνοντας πώς αυτές οι τεχνολογίες συγκρίνονται με τις παραδοσιακές πρακτικές μέσω βελτιωμένης λήψης αποφάσεων, προγνωστικής ανάλυσης και βελτιστοποίησης διαδικασιών. Η διατριβή στη συνέχεια στρέφει την προσοχή της σε μια συγκεκριμένη βιομηχανική διαδικασία, το θερμικό ψεκασμό μεγάλης ταχύτητας, μια τεχνολογία που ανήκει στο ευρύτερο πεδίο των επιφανειακών διεργασιών του κλάδου της τεχνολογίας υλικών. Αυτός ο τομέας, σημαντικός για το ρόλο του στην ενίσχυση των προϊόντων και τη λειτουργικότητα, βρίσκεται στην αρχή μιας μεγάλης αλλαγής χάρη στην ψηφιακή μεταμόρφωση που ωθείται από την TN και τις Βιομηχανικές Ψηφιακές Τεχνολογίες. Η μελέτη ανασκοπεί τα υπάρχοντα ψηφιακά εργαλεία, ταυτίζει κρίσιμες προκλήσεις και προτείνει πρωτοποριακές λύσεις βασισμένες στη μηχανική μάθηση για τη βελτιστοποίηση της απόθεσης επικαλύψεων, του ελέγχου παραγωγής και της παρακολούθησης των συστημάτων ψεκασμού κατά τη παραγωγή επικαλύψεων. Μια σημαντική ενότητα της διατριβής αφιερώνεται στην εφαρμογή προηγμένων μεθοδολογιών, όπως τα συνελκτικά νευρωνικά δίκτυα. Οι προτεινόμενες προσεγγίσεις αναδεικνύουν τις ανεξερεύνητες εφαρμογές της TN και σε συμπερασματικό επίπεδο, η διατριβή επικυρώνει αυτές τις καινοτόμες προσεγγίσεις της TN. Παρέχοντας τους κώδικες python για την εφαρμογή τους, η διατριβή αυτή προσπαθεί γεφυρώσει αποτελεσματικά το χάσμα μεταξύ θεωρητικής καινοτομίας και πρακτικής εφαρμογής. Θέλουμε να πιστεύουμε πως αυτή η εκτενής μελέτη συμβάλλει σημαντικά στον τομέα της βελτιστοποίησης βιομηχανικών διαδικασιών με τη χρήση τεχνητής νοημοσύνης, θέτοντας νέα πρότυπα και ανοίγοντας νέους δρόμους για έρευνα και εφαρμογή.

Λέξεις-κλειδιά: Τεχνητή Νοημοσύνη, Μηχανική Μάθηση, Συνελκτικά Νευρωνικά Δίκτυα, Θερμικός Ψεκασμός

List of Acronyms

AI	Artificial Intelligence
AM	Additive Manufacturing
ANN	Artificial Neural Network
ANOVA	Analysis of Variance
ASTM	American Society for Testing and Materials
CAT	computed axial tomography
CCD	charged-coupled device
CNN	Convolution Neural Networks
CS	Cold Spray
CSAM	Cold Spray Additive Manufacturing
DCF	Distance Correlation Functions
DOE	Design of Experiment
E2E	Enterprise to Enterprise
EBSD	Electron Back Scatter Diffraction
ECG	Electrocardiogram
EDS	Energy Dispersive X-Ray Spectroscopy
ESPCN	Efficient Sub-Pixel Convolution Network
FIB	Focused Ion Beam
HCP	Hard Chrome Plating
HEA	High Entropy Alloys
HVAF	High Velocity Air Fuel
HVOF	High Velocity Oxygen Fuel Thermal Spray
IDT	Industrial Digital Technologies

IIOT	Industrial Internet of Things
ISO	International Organization for Standardization
LDA	Laser Droplet Anemometry
MFC	Mass Flow Controllers
ML	Machine Learning
MSE	Mean Squared Error
OEM	Original Equipment Manufacturer
PLC	Programmable Logic Controller
PSNR	Pick Signal to Noise Ratio
QA	Quality Assurance
QC	Quality Control
QS	Quality Scheme
R&D	Research and Development
RELU	Rectified Linear Unit
RMS	Root Mean Square
RNN	Recursive Neural Networks
RSM	Response Surface Methodology
SEAC	Surface Engineering Advanced Coatings
SEM	Scanning Electron Microscopy
SMAW	Shielded metal arc welding
SOM	Self-Organizing Maps
SSIM	Structural Similarity Index Measure
SVM	Support Vector Machine
TCM	Tool Conditioning Monitoring

TS	Thermal Spray
VPPAW	Variable Polarity Plasma Arc Welding
WEF	World Economic Forum
XRD	X-Ray Diffraction

Contents

Chapter 1 Introduction.....	13
Contributions of this work	18
Chapter 2 Literature Review	21
2.1 Coating Deposition Modelling.....	21
2.1.1 Technology State-of-the-art	22
2.1.2 Machine Learning Prospects	25
2.2 Process Optimisation, Monitoring & Control.....	29
2.2.1 Challenges	29
2.2.2 Technology State-of-the-art Status	30
2.2.3 Overview of Machine Learning Approaches	33
2.3. Computer Vision and Machine Learning for Microstructural Characterization, Analysis and Standardisation	39
2.3.1 Challenges	39
2.3.2 Machine Learning Prospects	40
2.3.3 Coating microstructure similarity index as a conformance and quality metric	41
2.3.4 Microstructure Resolution Enhancement using Machine Learning.....	43
2.4 Improvement in thermal spray processing using Simulation, in process monitoring and post spray processing of data	45
Chapter 3 Process Monitoring and Control using CNNs	47
3.1 Methodology Audio and Image	47
3.2 Results and Discussion	48
Chapter 4. Process Monitoring and Control using YOLO	57
4.1 Methodology.....	57
4.2 Results & Discussions	59
4.2.1 Proposed Implementation Strategy and contributions of this work.....	59
4.2.2 Torch Fluid Dynamic Simulations	62
4.2.3 Classification based on Acoustic Emissions.....	64
4.2.4 Object Detection & Classification.....	67
4.2.5 Discussions and results	69
4.2.6 Industrial Use & Limitations.....	77
4.2.7 Conclusions	81
Chapter 5. Post Spray Quality Control and Microstructural analysis	83
5.1 Methodology.....	83
5.2 Results & Discussions of Post Spray Quality Control.....	85

5.2.1 Image similarity index	85
5.2.2 Cross sectional image classification	87
5.2.3 Object detection in cross sectional images.....	90
5.3 Results & Discussions of Microstructural Analysis Accuracy and Improvements	95
Chapter 6 Conclusions	98
List of References	102
Appendix A: Audio Classification Using CNNs.....	115
Appendix B: Spray Process Image Classification Using CNNs.....	121
Appendix C: Image Similarity Implementation Code	127
Appendix D: Coating Microstructure Defect Classification Code	128
Appendix E: Super Resolution Training Model Implementation Code.....	134

Chapter 1 Introduction

This work represents a comprehensive study that delves deep into the role of digital technologies, particularly Artificial Intelligence (AI) and Machine Learning (ML), in revolutionizing Surface Engineering and Advanced Coatings (SEAC) technologies.

Surface engineering is a critical field in modern manufacturing and industrial processes, dedicated to improving the surface properties of materials to enhance their performance, durability, and longevity. It involves various techniques and processes, one of the most prominent being thermal spray, which includes High Velocity Oxy-Fuel (HVOF) and Cold Spray processes. These advanced techniques play a vital role in a wide range of industries, from aerospace to automotive, and from medical devices to infrastructure.

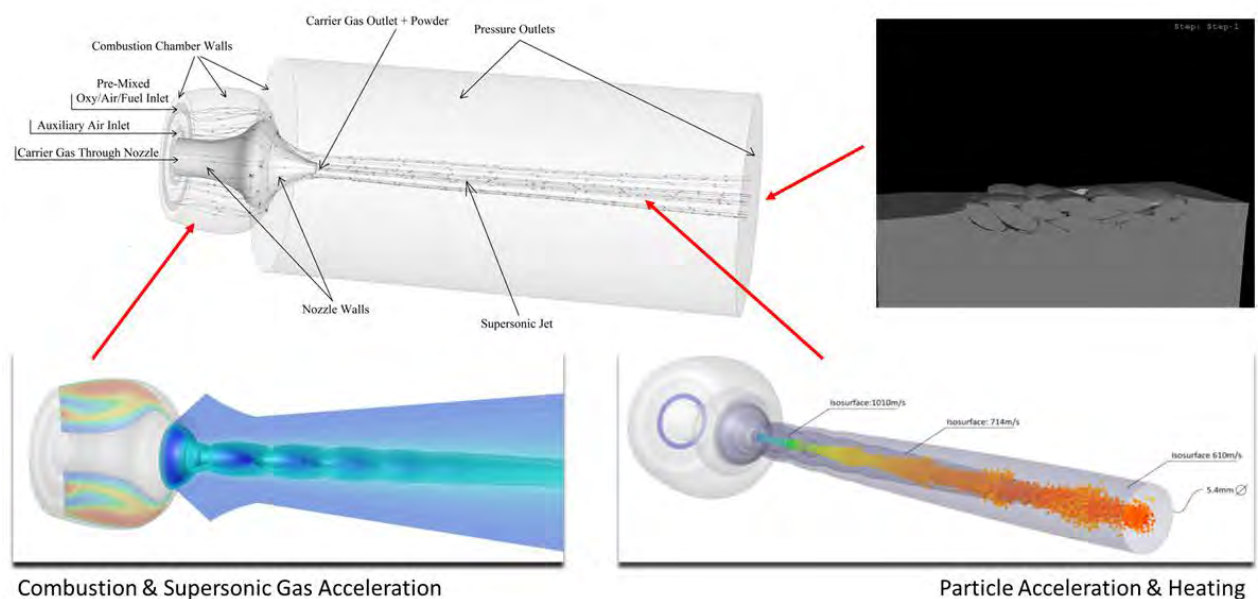


Figure 1.1 High Velocity Oxy-Fuel Thermal Spray Process.

HVOF is a thermal spray process that utilizes a combination of high velocity and combustion to apply coatings to a substrate. In HVOF (Figure 1.1 and 1.2), a mixture of fuel and oxygen is ignited in a combustion chamber, creating a high-temperature, high-velocity gas stream.

Powdered coating material is then injected into this stream, where it melts and is propelled towards the target surface. Upon impact, the molten droplets flatten and rapidly cool, forming a dense, strong coating. HVOF is renowned for producing coatings with excellent bond strength, low porosity, and high wear resistance. It is particularly effective for applying hard and wear-resistant materials like carbides and metals, making it ideal for components that are subjected to extreme wear and corrosive environments.

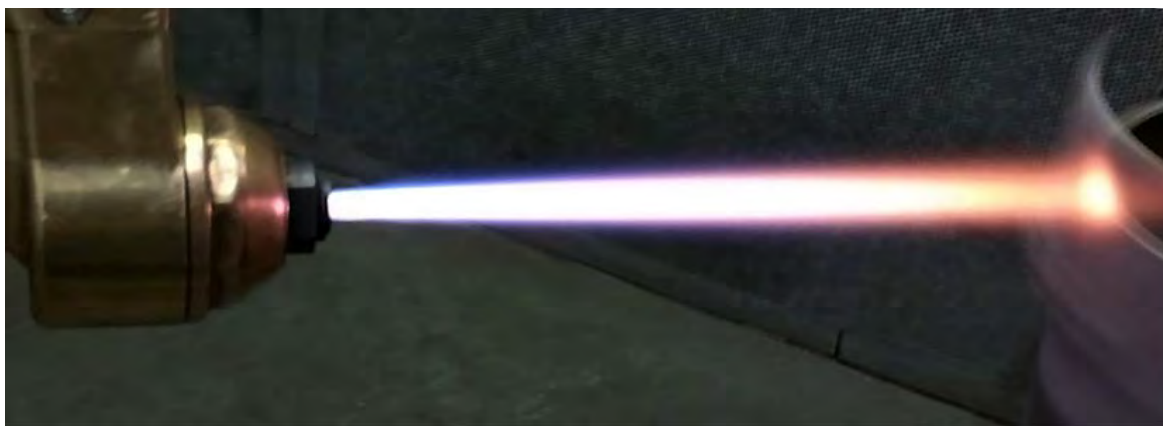
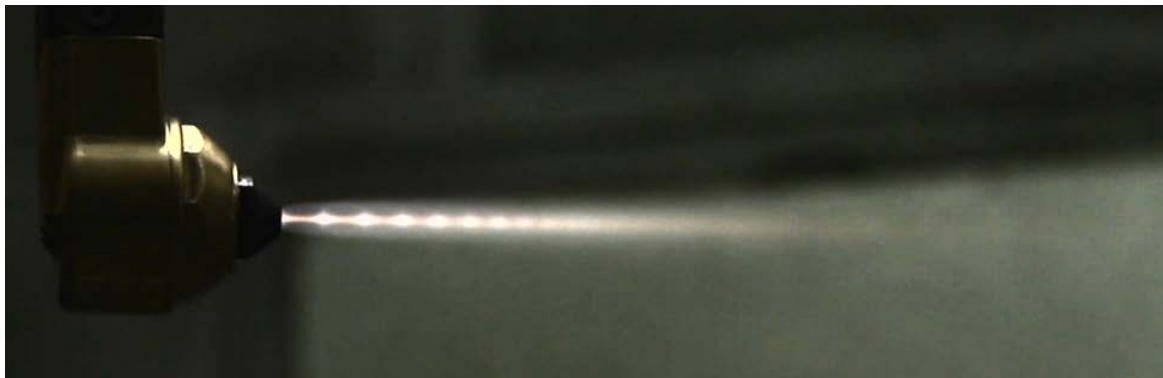


Figure 1.2 HVOF process. Top image shows the supersonic flame and bottom the coating deposition where metal powder is accelerated and heated towards a substrate where the coating forms.

Cold Spray, on the other hand, is a process where particles are accelerated to high velocities at temperatures that are relatively low compared to other thermal spray processes. In Cold Spray, a high-speed gas jet is used to accelerate metal particles towards a substrate. Unlike HVOF, the particles do not melt in the gas stream; instead, they deform plastically upon impact,

bonding to the surface and to each other to form a coating. This low-temperature process minimizes oxidation and retains the original properties of the feedstock material, making it suitable for materials that are sensitive to high temperatures. Cold Spray is increasingly used for repair applications, corrosion-resistant coatings, and for manufacturing parts with unique material properties.

Both HVOF and Cold Spray processes have their distinct advantages and are chosen based on the specific requirements of the application, such as the desired properties of the coating, the nature of the substrate material, and the operating environment of the coated component. The development of these technologies is closely tied to advancements in surface engineering, as they offer innovative solutions for enhancing material properties and extending the life and performance of critical components in various industries.

The increasing global market competition and changing customer demands necessitate that SEAC technologies not only maintain high quality and low costs but also adapt rapidly to different production scales. The utilization of digital technology, especially in thermal spray processes, is becoming indispensable for achieving this flexibility, optimizing quality management, and enhancing supply chain efficiency.

The thesis underscores the significance of Industrial Digital Technologies (IDTs) in enabling a thermal spray shop to monitor the health of its equipment and processes proactively. This proactive approach is critical for predicting and rectifying issues before they impact production, thus increasing asset availability. The increasing transparency and global connectivity in technology sharing create a highly interdependent ecosystem within the surface engineering sector. In such a landscape, the lack of sophisticated digital technologies can drastically affect the quality of technical and service support, leading to disruptions in the supply chain, potential warranty and guarantee commitments, and even loss of contracts.

The emergence of IDTs as a solution to these challenges is evident, with a 2017 World Economic Forum report highlighting a \$100 trillion opportunity for society and industry by 2025. The COVID-19 crisis has further emphasized the need for a holistic approach that leverages the Industrial Internet of Things (IIoT) for SEAC technologies. However, despite advancements in other sectors, most thermal spray equipment manufacturers and spray shops are currently limited to using sensors at the component level. There is a pressing need for AI-driven technologies that address higher system and process levels, such as advanced thermal spray simulation packages for real-time, non-destructive coating quality monitoring, machine vision technologies for data-driven asset condition monitoring, and intelligent coating quality inspection and characterization.

The production steps in a typical thermal spray shop are divided into three types of operations:

1. Assessing the complexity of a new job through modelling and simulations,
2. Effective process monitoring and control during spraying, and
3. Post-spray quality checks.

Figure 1.3 presents a comprehensive experimental setup for a thermal spray coating process, featuring an array of specialized equipment and sensors integrated to ensure precision, safety, and monitoring. Central to the setup is a robotic arm equipped with a torch at its end, which is responsible for the application of the coating material. The robotic arm is designed with an extension arm and gun rotation capabilities, allowing for a wide range of motion and angles for the spray, which is critical for uniform coating applications over complex geometries. The torch nozzle, visible in a close-up view, is where the material is expelled at high temperatures to create the coating.

Adjacent to the robotic arm are acoustic sensors, strategically placed to monitor the process by capturing airborne acoustic emissions, providing real-time data for process control and quality assurance. This is an innovative approach that supplements traditional monitoring techniques and offers the potential for non-invasive process diagnostics. Safety and operation are further overseen by gas sensors and interlocks, essential components that detect gas levels and ensure that the system operates within safe parameters, automatically shutting down the process if needed to prevent accidents.

Thermal imaging equipment is also part of the setup, which plays a crucial role in monitoring the temperature profile during the spraying process, ensuring that the material properties are consistent and within specifications. Additionally, a digital control panel is in place for the precise regulation of gas and water flow, which is integral to maintaining the correct mixture and pressure for the thermal spray process.

Figure 1.3 captures a current state-of-the-art thermal spray coating system, illustrating the technology and complex integration of various control and monitoring mechanisms that together, contribute to the efficient, safe, and high-quality application of thermal coatings in an industrial context.

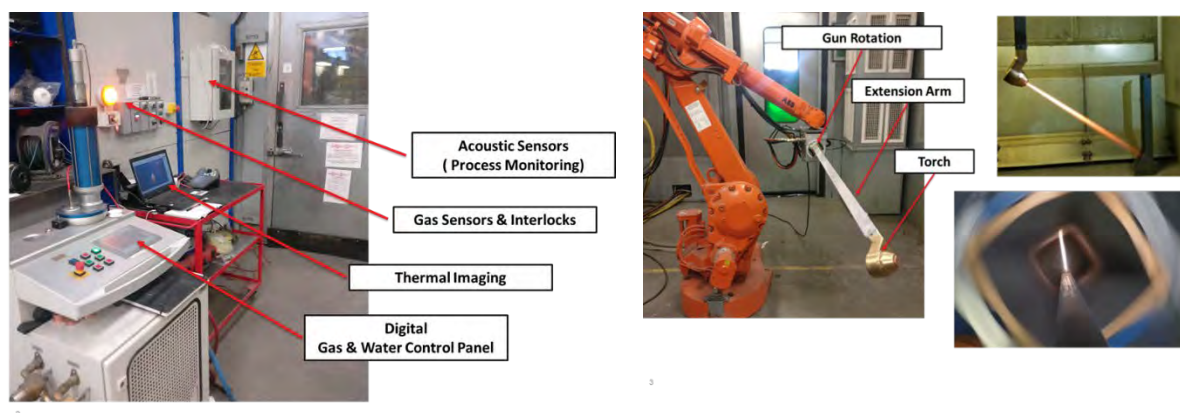


Figure 1.3 Experimental setup within a thermal spray booth, used to collecting data for machine learning (ML) training.

Contributions of this work

The contributions and impacts of this work in the realm of thermal spray technologies are both profound and multifaceted, addressing critical gaps identified through an extensive literature review and offering innovative solutions that promise to revolutionize both process monitoring and quality control in industrial settings.

One of the most significant challenges in thermal spray processes highlighted by the literature review is the lack of flexible, cost-effective, and user-friendly monitoring tools that can operate effectively in the dynamic and complex environment of industrial applications. Traditional monitoring equipment, while sophisticated and capable of providing detailed data, is often expensive, static, and requires specific conditions to operate. These limitations severely restrict their practical utility in industrial contexts, where the spray process is dynamic, and the spray torch is constantly in motion.

The work presented develops a novel system designed to overcome these challenges, utilizing advanced Convolutional Neural Network (CNN) architectures to monitor the thermal spray process in real time. This system represents a significant leap forward in process monitoring technology, offering a solution that is not only more adaptable and easier to integrate into existing workflows but also more cost-effective than traditional academic instruments. By demonstrating the efficacy of this system in accurately capturing and analyzing the dynamic aspects of the spray process, the author makes a compelling case for its broad adoption in the industry, potentially leading to significant improvements in process control, efficiency, and overall quality of the coatings produced.

At the quality control stage, the need for reliable, repeatable measurements of coating characteristics is paramount. Current industry practices predominantly rely on optical microscopy and manual metallographic examination, processes that are not only time-

consuming and costly but also subject to variability and limitations in the resolution and clarity of the images obtained. Additionally, the absence of established metrics for assessing the similarity of coatings further complicates the quality control process, making it difficult to ensure consistent quality across different batches of coatings.

In response to these challenges, the author has developed three key tools that collectively represent a groundbreaking advancement in coating quality assessment. First, the application of super-resolution techniques to optical microscope images significantly enhances image quality, enabling more precise and accurate analysis of coating characteristics. Second, the introduction of an image similarity index provides a novel metric for objectively comparing coatings, addressing the longstanding gap in quality control measures. Lastly, the development of a CNN model capable of classifying coatings based on their metallographic characteristics offers a powerful tool for automating the quality control process. These tools not only promise to improve the reliability and repeatability of quality assessments but also have the potential to reduce the costs and labour associated with manual examinations, making high-quality control more accessible and feasible for a broader range of industrial applications.

The impact of this work extends beyond the immediate advancements in process monitoring and quality control technologies. By addressing critical gaps in the current state of thermal spray technology with innovative ML and computer vision solutions, this work not only enhances the efficiency and quality of thermal spray processes but also paves the way for further research and development in this field. The successful implementation of these tools can lead to broader acceptance and adoption of ML techniques in surface engineering, driving innovation and improvement across the industry. Furthermore, the practical applications of this research—ranging from improved process control to more accurate and reliable quality control—have the potential to significantly increase productivity, reduce costs, and elevate the

standard of quality within the thermal spray community, contributing to the advancement of surface engineering as a whole.

This work has been published in journals and presented at international conferences, and has already garnered citations in acknowledgment of its significance:

1. Digital transformation of thermal and cold spray processes with emphasis on machine learning, **K Malamousi**, K Delibasis, B Allcock, S Kamnis, Surface and Coatings Technology 433, 128138, 2022.
2. Roadmap on signal processing for next generation measurement systems, DK Iakovidis, **K. Malamousi** et.al. Measurement Science and Technology 33 (1), 012002,18,2021
3. Aeroacoustics and artificial neural network modeling of airborne acoustic emissions during high kinetic energy thermal spraying, S Kamnis, **K Malamousi**, A Marrs, B Allcock, K Delibasis, Journal of Thermal Spray Technology 28, 946-962.
4. Real-Time Thermal Spray Process Monitoring Using Convolution Neural Network Deep Learning Architectures, **K. Malamousi**, K. Delibasis, S. Kamnis, J Therm Spray Tech, <https://doi.org/10.1007/s11666-024-01713-7>.

Chapter 2 Literature Review

2.1 Coating Deposition Modelling

Thermal and cold spray processes provide a transformative opportunity to apply materials on the surface of components that are tailored to the distinct needs of a given application, thereby shifting the paradigm from "*apply the material you have*" to "*engineer the material and process you need*". In this direction, and despite all intense efforts, only few of the innovative spray coating materials discovered in the lab have earned their market place in a reasonable timeframe. This is mainly due to great variations in the application method of same materials, the deposition kinematic effects in thermal and cold spray, the new complex part designs adopted by the original equipment manufacturers (OEM) and the lack of information related to the mode of operation of the coated components. Academic lab testing can provide the key material properties; however, it is difficult to know whether the conclusions reached can be scaled up to more complex routines invoked in new coating industrial application. High value manufacturing sectors are continually seeking new ways to improve the performance and durability of critical components such as those used in aerospace, defence and automotive sectors. As a result, multiple design iterations are required in an effort to converge to a part-coating system solution while extensive product qualifications come at high cost for both the OEM's and the supply chain.

Given the industry's transition away from hard chromium plating to more environmentally friendly alternatives, such as kinetic spray processes, the need for accurate and flexible coating simulation tools comes at an opportune time. Within the context of thermal and cold spraying, a pre-spray optimization strategy has to satisfy several requirements. The central requirement is minimizing the coating thickness variation, achieve uniform or tailored coating properties using a model which cannot be expected to be analytically available. The ability to evaluate

and visualise in real time the as-sprayed coating properties on difficult to reach areas of intricate components is of paramount importance and an issue that industrialists come across regularly at the qualification stage of a coating application. A further important requirement is the optimization of the kinematic quality of the robotic spray path, with respect for example to the surface curvature, shadowing and velocity. From a computational point of view, the number of evaluations of the simulation should be kept small and the software should be able to provide multiple realistic degrees of freedom. Preferably, from an industrialist's point of view, the predictive algorithm should be based on analytical topological models and experimentally driven inputs to minimize truncated numerical errors.

The ideal simulation platform should be able to satisfy the above-mentioned requirements and to bridge several technological gaps by offering a flexible and accurate computational tool that can map any measurable in the lab coating property (wear rate, porosity, microhardness, thickness etc.) on the surface of simulated parts using different materials, processes, kinematic conditions and operating parameters. The approach should offer effective integration into the product design cycle of original equipment and could be used as an R&D process optimisation tool that would allow fast technology transfer from the lab to production. Ultimately this technology should offer valuable material property information that could potentially lead to improved product designs at a substrate/coating system level.

2.1.1 Technology State-of-the-art

Several computer-based simulations of coating deposition processes have been published in an effort to predict the main parameters of thermal coating spraying process, such as the thickness. In a work by [1] a plasma coating numerical methodology has been described, that can handle objects with complex shapes. This technique uses the level set method to calculate the coating thickness and the approach appears to be promising with accurate results been demonstrated for one complicated axisymmetric object. Several other studies have focused on the

optimization of the coating deposition process. The majority of published work is structured around the development of particle impact models coupled with empirical best-practice rules.

In [2][3] the optimization of the nozzle mounting is proposed, using a simulation environment (RobotStudio™) [4]. More specifically, the kinematic parameters and the position of the robotic arm joints are considered to simulate simple planar plates. Another work [5] suggests a post optimization of the gun kinematics, using a deposition model to simulate the coating process. Given the specific gun trajectories and poses (position, direction), the nonlinear conjugate gradient method [6] is used for spray pattern optimisation. A similar work is reported in [7] where the gun trajectory is optimized for free form objects with simultaneous calculation of the heat and mass transfer. In [8] two methods of optimization of the speed and trajectory of the torch for specific coating paths are proposed. The trials take into consideration the difficulties occurred by the weight and complexity of the robotic arm needed to deposit the coating. These methods are verified by conducting simulations in RobotStudio™ and experiments.

Offline methods for programming a robotic torch in order to achieve the desirable result with the minimum error in the coating parameters are presented in [9], where the trajectory and the speed of the torch are optimized, using simulations before the actual process. In [10] the relationship between the coating profile, spray distance and scanning step is investigated and the RobotStudio™ programming suite is used to generate optimized robot trajectories for thermal and cold spray. A method is presented in [11] that predicts the thickness and profile of coated cylindrical surfaces as a function of the scanning step that is determined by the radius of curvature and scanning angle. Based on those parameters a robotic path for the coating is generated. The design of a new software package for the off-line programming software RobotStudio™ is proposed in [12] that generates automatically robot trajectories considering a relatively simple coating model that takes into account the kinematic parameters of the torch.

A numerical model for the heat transfer during plasma coating of cylindrical surfaces is also proposed in [13]. Several numerical methods are suggested for the simulation of splat formation during thermal and cold spray [14][15][16][17][18][19][20][21][22][23][24]. These methods provide critical information for understanding the underlying physics of the process but are less suitable for full scale coating deposition simulations.

In relation to cold spray (CS) the work by [25], presents a geodesic-based robot trajectory planning approach. The methodology is based on the Gauss–Bonnet theorem by means of curve discrete algorithm used to construct orthogonal planes and scanning curves creating target points for the robot trajectory. Another study [26] introduced a new approach for simulating the coating thickness in CS, especially for the prediction of shadow effects. The evolving geometric 3D model can offer realistic build-up morphologies and accurate profile predictions. The platform enables the integration of robot programming with simulation to better control the coating process. In its current form the model can predict only the coating thickness but, in the future, process physical phenomena could be implemented in order to provide other coating quality indicators such as porosity values. Several other publications are focusing on additive manufacturing through CS [27] [28][29].

Works by [30][31] present a different perspective on the development of spray coating simulation tools. In this approach MATLABTM [32] routines calculate the evolution of coating properties on the surface of sprayed parts from combined generic experimental and process specific calibration data in the form of high order interpolated lookup tables. In this study [31] the geometry is triangulated and the kinematic values of the projected spray plume are stored on each triangle barycentre. Then the equivalent coating property or multiple properties, stored in reference tables, are assigned to each triangulated surface element of the object under investigation. This simulation tool does not include automated robot path planning and is limited to manual trial and error process optimisation in a virtual environment. On the other

hand, the predictions are error-free and the calculations are performed analytically in almost real-time. With the integration of motion sensors, the technology is suitable for mapping the coating properties on the surface of the part during and post spray in a non-destructive manner. In Figure 2.1a the coating thickness distribution is shown for a rotating object sprayed with the torch moving in front of the part at constant traverse speed pointing towards its centre of rotation. In Figure 2.1b the coating microstructure is shown as a contour plot on the surface of a complex rotor geometry.

2.1.2 Machine Learning Prospects

Among numerous neural network architectures that could be used to address the robot path planning challenges, one is of particular interest and was introduced by Finish Professor Teuvo Kohonen in the 1980s, [33]. Self-organizing map (SOM), sometimes also called a Kohonen map use unsupervised, competitive learning methods producing low dimensional, discretized representation of presented high dimensional data, while simultaneously preserving similarity relations between the presented data items. Such low dimensional representation is called a feature map. Despite its simplicity, SOMs have been used historically in various applications. This in a broad sense includes visualizations, generation of feature maps, pattern recognition and classification. Some applications of SOMs focus on control of robot arm, learning motion maps,

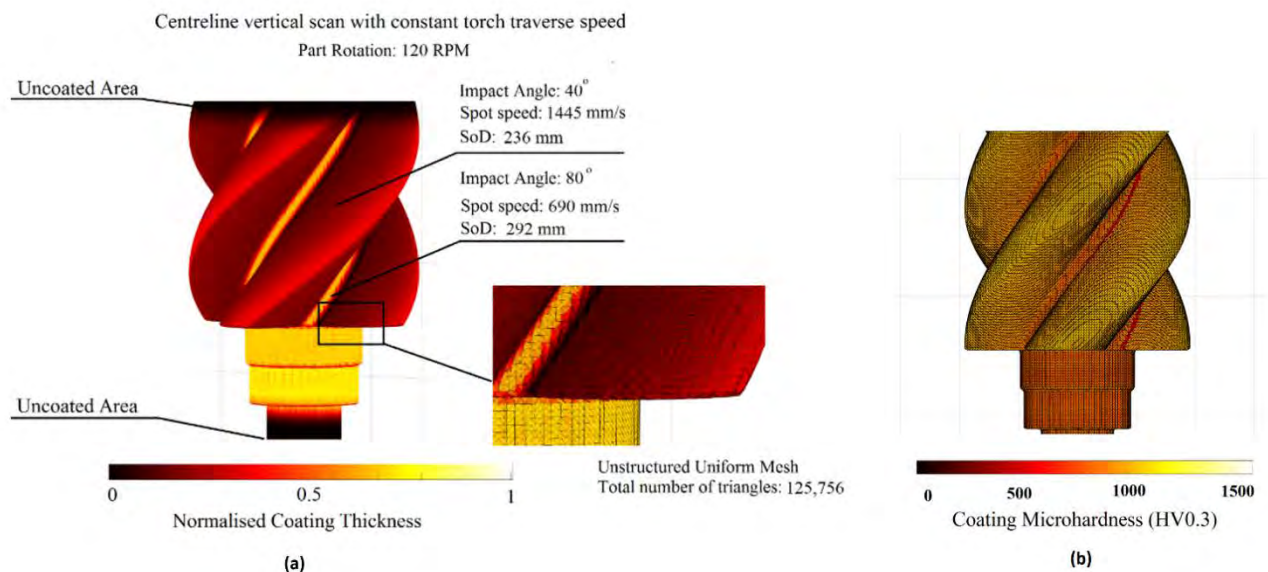


Figure 2.1 (a) Simulated coating thickness evolution on a complex industrial part: Oil & Gas mud rotor HVOF thermal spraying [31], (b) Simulated coating microhardness

collision avoidance for multi-vehicle systems including navigation and robotics. [34][35][36]. SOMs have found applications in chemistry [37], recognition of diseases from medical images (ECG, CAT scans, ultrasonic scans, etc.), similarity of music recordings [38], maritime applications for the analysis of passive sonar recordings and for planning ship trajectories [39], classification of satellite images [40] and many other. A SOM is a single layer neural network with units set along an n -dimensional grid. Most applications use two-dimensional and rectangular grid, although many applications also use hexagonal grids, and someone, three, or more dimensional spaces. SOMs produce low-dimensional projection images of high-dimensional data distributions, in which the similarity relations between the data items are preserved.

In relation to cold and other thermal spray applications unsupervised algorithms based on SOMs can be used for the configuration of an optimal robotic spray planning, for any given object. A detailed description of this approach can be found in this published work [41]. The SOM is defined as an ordered sequence of nodes (or neurons). The nodes represent the

consecutive positions of the spraying gun as it moves along its trajectory with variable speed and jet direction at each position. Each node contains its position, the direction of the jet and the speed magnitude with direction from the current node to the next one. For a given substrate, the SOM is initialized with the position of the nodes equally spaced in a full circle, clockwise, around the centre of mass of the object. For every repetition the triangles visible from at least one node are stored in a table and a random triangle is selected from this table. A winner node and its neighbours are then calculated. The weights (position, speed and direction of the beam) of these nodes are appropriately updated so that the coating property of interest approaches the optimal value.

The algorithm has been tested and compared against current best practise for a relatively complex geometry that is often processed in thermal spray shops. This is a mud rotor used in oil and gas drilling application. The coating achieved by the proposed algorithm for the mud rotor is shown in Figure 2.2 at the plane indicated in Figure 2.2a. The coating thickness along the object intersection marked in yellow is plotted for the SOM based method and for the expert in Figure 2.2b and c and respectively. It can be visually observed that the thickness achieved by the SOM method is far more uniform than the one achieved by current spray best practices. The benefits of using this method are profound as the complexity of the geometry to be sprayed increases.

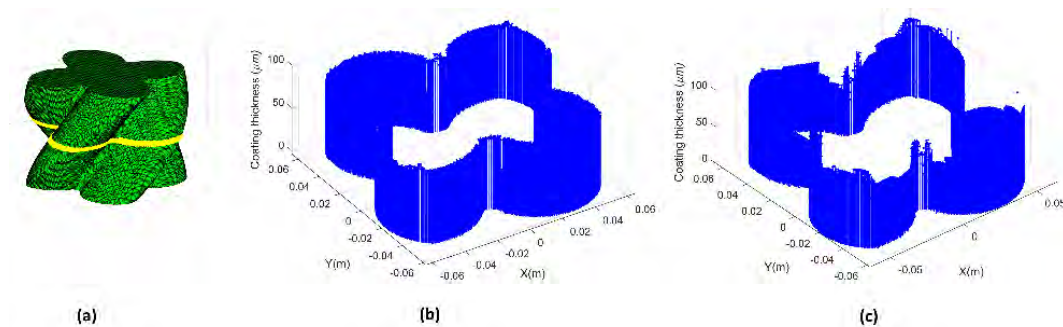


Figure 2.2. (a) Coating plane of the triangulated mud rotor. Coating thickness of the mud rotor at the selected plane using the (b) SOM method [41] and (c) Simulation optimised method [31]

ANNs have been also used in Cold Spray Additive Manufacturing (CSAM) processes for profile modelling. These techniques form the basis for modelling full scale topologies combined with overlapping and overlay approaches [42][43]. The two main single-track profile modelling approaches use two distinct methods: mathematical and data-driven. In wire arc additive manufacturing the mathematical functions have been combined with regression models to provide predictive capabilities. In [44] a quadratic regression model was used based on experimental data to compute the coefficients of the parabolic function model for a single-track profile. In cold spray additive manufacturing, a mathematical Gaussian function model is often selected[45][46] but also other functions have been used including triangular [47][48] and trapezoidal [49] forms. Despite the relatively large number of approaches reported in the literature, there still no agreement on a universal mathematical function that is suitable for a wide range of process conditions. To address this challenge, data-driven modelling has emerged as promising alternative due to its exceptional non-linear mapping capability especially when combined with process specific features and kernel design.

Comparing the results achieved from ANN approaches with those from quadratic regression model [50], it is evident that the ANN model can outperform the pure mathematical approaches in the prediction of key geometric features. Nevertheless, data-driven modelling has previously been limited to only predicting the height and width of a single track and not the entire profile. In this direction [51] adopted an ANN modelling approach for cold spray applications placing more emphasis on the morphology of a single track at normal and off-normal spray angles. This work demonstrated the potential of a data-driven modelling approach for better prediction accuracy than the Gaussian function model. Despite the reasonable prediction accuracy achieved with ANN methods, several studies [52][53] revealed the existence of more complex and nonlinear phenomena requiring further exploration in order to improve the prediction accuracy and generalisation. The necessity for larger datasets and the associated high costs to

obtain them was the main drive for combining ANN and mathematical modelling approaches. A representative example of the combinatorial approach is presented in [54]. In this work the authors present the application of a data-driven modelling approach with a data efficient focus in the prediction of a single-track profile in cold spray additive manufacturing.

This result demonstrates that a data driven modelling approach can outperform a conventionally used mathematical function model in cold spray additive manufacturing, both at normal and off-normal spray angles, with appropriate data efficient modelling techniques. Moreover, these techniques harnessed the existing model in developing a new data-driven ANN model without further experimentation.

2.2 Process Optimisation, Monitoring & Control

2.2.1 Challenges

As the regulations concerning hard chrome plating (HCP) are getting more stringent and the variety of applications that benefit from thermal spray (TS) and cold spray coatings is expanded, more sophisticated ways of monitoring and controlling the quality and process efficiency are urgently needed. At the same time, there is an increasing interest in low temperature TS methods in order to avoid unwanted reactions in the deposited materials that occur in higher temperatures. The family of WC-Co hard metal coatings deposited via HVOF or HVAF outperform HCP and is the driving force behind this shift to lower deposition temperatures because they have established themselves as the leading successor of HCP [55][56]. However, in these temperatures, particles are often solid or partially molten and thus, rely primarily on their kinetic energy to adhere and form a dense coating. Furthermore, the broad particles size distribution of the used powders and variations in spatial intensity and temperature distribution of the flame emission lead to an accordingly broad range of velocity and temperature values at the moment of impact for individual particles. Usually, larger

particles arrive at the substrate solid and with insufficient kinetic energy due to slower acceleration in the gas stream and bounce off (elastic impact). On the contrary, smaller particles impact the substrate melted due to better heat exchange with the flame, not contributing to the needed compressive residual stresses in the coating and causing porosity.

The above conditions create a sensitive balance that depends on a lot of process parameters in order to achieve optimum deposition for the majority of the particles of the used powder. Advanced control systems, based on mass flow and programmable logic controllers, ~~those~~ used with thermal spraying equipment, lead to reproducible primary input parameters (gas flow rates, pressures, feed rates, gun movement), but they are not technically capable of monitoring the variation of the coating quality resulting from permanent or sudden modifications and fluctuations in the system hardware, such as nozzle wear, spitting, troubling in feeding lines, gas leaks, tube erosion, etc., undetectable by these systems but extremely important for practical applications. These modifications in the spraying parameters can produce great differences in the quality of the coating that are not detectable during spray. Such differences in the quality can produce catastrophic failures of in-service components. In Europe there are more than 3000 mid-sized and small companies that produce thermal sprayed coatings with a total turnover estimated to almost 1 billion euros. Companies have big interest in guarantying the quality of the coatings they are producing to fulfill the rising technical requirements demanded by the end-users. The best way to assure the quality of the coatings in a direct way is to measure continuously the “characteristics” of the process and to record all fluctuations.

2.2.2 Technology State-of-the-art Status

Many different approaches have been presented throughout the years in an effort to address the process monitoring and control challenge as presented in [57] and initiated in 1991 by C.Moreau et.al who filled a patent [58] providing the basis of sensing technologies. A thorough

review of sensors used to control and monitor thermal spray processes is presented in [59]. Several devices have been used during the early days of thermal spray technologies such as the laser Doppler anemometry (LDA) for particle velocities, phase Doppler shift to also measure their diameters, fast pyrometers for temperature measurements and CCD cameras for particle trajectory tracking [60][61]. A decade later studies by [62][63] formed the basis for the development of the first two commercial condition monitoring systems.

The DPV-2000 System (Tecnar Automation, Quebec, CN) and the Spray Watch by Oseir and later on the Accuspray system by Tecnar. These devices were robust enough to work in the harsh environment of spray booths. They allowed monitoring the effect of the conventional spray parameters (gas flow rates, nozzle internal diameter, power level for plasma spray torches, injection conditions, particles size distribution and morphologies...) onto the in-flight particle parameters. Several closed-loop control methods have been proposed [64][65][66][67] in an effort to develop real-time process diagnostics using the aforementioned monitoring technologies and many other works devoted to the linkage between in-flight particle measurements and coating properties [68][69][70][71][72][73][74][75].

In the same direction, research into acoustic emissions from the thermal spray process has primarily focused on the use of sensors in contact with the sprayed substrate to monitor the kinetic properties of particle deposition and the growth of coating defects such as cracks and delamination. Successful experiments using a contact sensor were conducted by [76] to measure the total kinetic energy of semi-molten and solid particles impacting the substrate. Several other works [77][78][79] studied the coating health status using acoustic emissions mainly focusing on crack formation and propagation. Non-contact laser Acoustic Emission (AE) techniques to monitor the formation of defects in coatings were used by [80]. These methods have shown promise in accurately monitoring the quality of thermal spray coatings as they are being sprayed, but the contact sensor must be carefully positioned on a stationary

substrate and calibrated for every part being sprayed. This drastically increases spraying time for each part and would necessitate training operators in the use of the monitoring equipment. These studies rely on the Rayleigh waves on the substrate surface due to particle impact. Airborne acoustic emission studies, instead of contact sensors, for thermal spray process monitoring are very limited; However, in other processes such as various welding processes (VPPAW, SMAW, laser welding) and machining processes, airborne acoustic emissions are closely monitored and analysed to give real-time predictions about certain aspects of the outcome of the process, or health of the hardware in use [81][82][83][84][85].

The increased number of process data from various sensors throughout the years was essential for the development of data modelling methodologies using statistical, machine learning and other approaches. The designs of experiments (DOE) methodology useful for thermal spraying and associated processes of post-spray treatment are thoroughly reviewed in [86] with examples of the application of different designs to determine the response equations with the responses related to microstructure, mechanical, electrical and other properties of coatings deposited using different thermal spray and post-spray processes are presented. The Hadamard, two-level fractional and Taguchi design of experiments with the two-level factors used in thermal spraying are presented in [87]. When the relationship between the experimental response and factors is not linear the response surface methodology (RSM) is used [88][89].

Methods such as the analysis of variance (ANOVA) for determining the significance of the regression coefficients (experimental factors) are presented in [90][91][92][93]. Many other studies are also reported on the application of statistical techniques in surface coating methods [94][95][96][97]. Even though statistical techniques are highly efficient in surface coating process modeling they have also shown some restrictions in handling large number of process parameters and in turn obtaining a judicious relationship between input and output data. Such

difficulties were observed mainly due to the nonlinear and complex nature of surface coating methods and parameters involved in the process.

The extensive use of sensors and data analytics, primarily as academic and R&D instruments, have allowed a much better understanding of the influence of the spray gun working conditions on the particles in-flight parameters, especially their temperatures and velocities. As a result, industrialists have achieved better process reproducibility and reliability but most importantly these technologies have contributed significantly towards the development of new coating applications through material deposition optimization. Nevertheless, there are several key limiting factors that have prevented the technology uptake.

The condition monitoring systems are expensive and their return on investment cannot be easily justified particularly when small and medium size spray shops are concerned. The three main commercially available systems are more suitable for static monitoring. It is difficult to mount them on a spray torch for dynamic monitoring while spraying. When spraying complex parts, the working envelope of the torch is significantly reduced due to compromised angles and close proximity manipulations. The dust in the booth from bouncing back particles, the high booth temperature during prolonged spray operations and many other practical reasons when spraying internal surfaces render these technologies impractical for everyday use in a thermal spray shop. Ideally the conditioning unit and its integrated sensors should be compact, easy to use, robust and able to capture process related data that could be linked to coating properties in-real time with minimum latency.

2.2.3 Overview of Machine Learning Approaches

In this direction, machine learning (ML) techniques with the ability to solve intricate and highly nonlinear processes have attracted worldwide researchers to use them for process optimization, monitoring and control. Different machine learning techniques such as Artificial Neural

Networks (ANN) and Convolution Neural Networks (CNN) have been used extensively due to their capability of learning from the past data and using them to predict the target variable more accurately. It is out of this review's scope to outline the theory behind machine learning thus only a brief introduction is provided here.

The Artificial Neural Network is a learning algorithm that emerged and evolved from the idea of Biological Neural Networks of human brains [98]. ANN works very similar to the biological neural networks but doesn't exactly resemble its workings. ANN algorithm would accept only numeric and structured data as input. To accept unstructured and non-numeric data formats such as Image, Text, and Acoustics, Convolutional Neural Networks (CNN) [99], and Recursive Neural Networks (RNN) [100] are used respectively. ANNs consists of two phases the forward propagation and backpropagation. The forward propagation involves multiplying weights, adding bias, and applying activation functions to the inputs and propagating it forward. The purpose of an activation function is to introduce non-linearity to the data. Introducing non-linearity helps to identify the underlying patterns which are complex. The three common activation functions are the Logistic/ Sigmoid, Hyperbolic Tangent and the Rectified Linear Unit (ReLU) [101]. The backpropagation step is the most important step which usually involves finding optimal parameters for the model by propagating in the backward direction of the Neural network layers. The backpropagation requires an optimization function to find the optimal weights for the model. The main optimization functions available are the Gradient Descent[102], Adam optimizer [103], Gradient Descent with momentum[104] and RMS Prop (Root Mean Square Prop) [105].

The Neural Architecture consists of single or multiple layers including input layers, output layers, and hidden layers. An example of a Convolutional Neural Network (CNN) architecture for image classification is shown in figure 2.3.

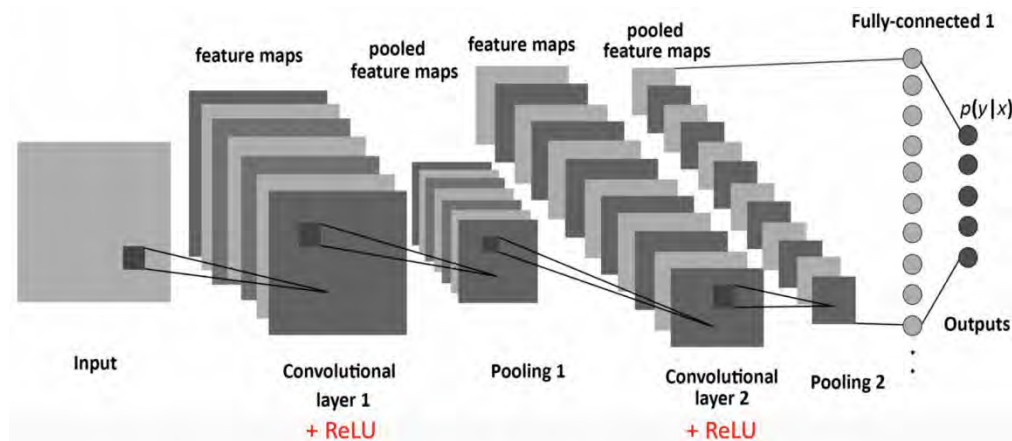


Figure 2.3. An example CNN with two convolutional layers, two pooling layers, and a fully connected layer which decides the final classification of the image into one of several categories[106]

The basic functionality of the example CNN in Figure 2.3 can be broken down into four key areas. As found in other forms of ANN, the input layer will hold the pixel values of the image. The convolutional layer will determine the output of neurons of which are connected to local regions of the input through the calculation of the scalar product between their weights and the region connected to the input volume. The rectified linear unit (ReLU) aims to apply an 'elementwise' activation function such as sigmoid to the output of the activation produced by the previous layer. The pooling layer will then simply perform down sampling along the spatial dimensionality of the given input, further reducing the number of parameters within that activation. Finally, the fully connected layers will perform the same duties found in standard ANNs and attempt to produce class scores from the activations, to be used for classification. Through this simple method of transformation, CNNs are able to transform the original input layer by layer using convolutional and down sampling techniques to produce class scores for classification and regression purposes.

An interesting study by [107] proposes a decision support model by employing convolutional neural networks for the optimization of control parameters in plasma spraying processes. In this approach, process simulation data are used as input values instead of actual experimental data. The results demonstrate potential generalization performance of the CNN models for

developing spray recipes to achieve the desired coating property. Despite the powerful features and huge surge in CNN popularity after 2012 there are only few published studies in the field of thermal spray. In the following sections, several examples with implementation codes are presented in an effort to showcase the industrial applicability and effectiveness of CNNs.

On the contrary, standard and hybrid ANNs have been extensively used to study the thermal spray process and the resulting coating properties. One of the first studies back in 2002 [108] developed a generalized ANN approach and presented implementation protocols in relation to plasma spray process. In relation to process control, diagnostic sensors and predictive systems are coupled in [109] for a plasma spray process. The training dataset was constructed from recorded values using a DPV2000 (TECNAR Automation, St-Bruno, QC, Canada) optical sensing device. The adhesion strength of plasma sprayed coatings is examined in [110]. This study highlighted the influence of in-flight particle characteristics on bond strength and the results were quantitatively demonstrated and predicted with the use of ANNs.

In several other studies [111][112][113][114][115][116][117][118][119][120] optimized ANNs were developed and trained from experimental data for process optimization, control and coating property prediction purposes. More specifically, in [121] the wear data are used to create predictive ANN models for the material loss. In practice this approach can be used to develop accelerated wear testing procedures.

A different approach was presented in 2019 by [122] where ANNs are trained from HVOF airborne acoustic signal inputs and experimental coating property outputs. The main objective of this work was to demonstrate a methodology that could potentially lead to the design of a simple process monitoring device based on the airborne acoustic emissions generated during the HVOF or other high kinetic energy processes. The results demonstrate that intelligent sampling and QA monitoring linked to spray processing parameters is feasible using only a

microphone and an ANN inference model for data processing. The modeling data confirmed the presence of unique detectable noise features that can be attributed and correlated to several process variables. Most importantly, this work demonstrated for the first time that the information carried in the raw acoustic dataset contains process variable specific signatures that an ANN (Figure 2.4) can benefit from producing meaningful and accurate outputs in real-time. The generalization and predictive capability of the model has been successfully assessed and validated. From this work intelligent airborne noise sampling appears to be a cost-effective and more practical alternative to optical and other sensors during spray.

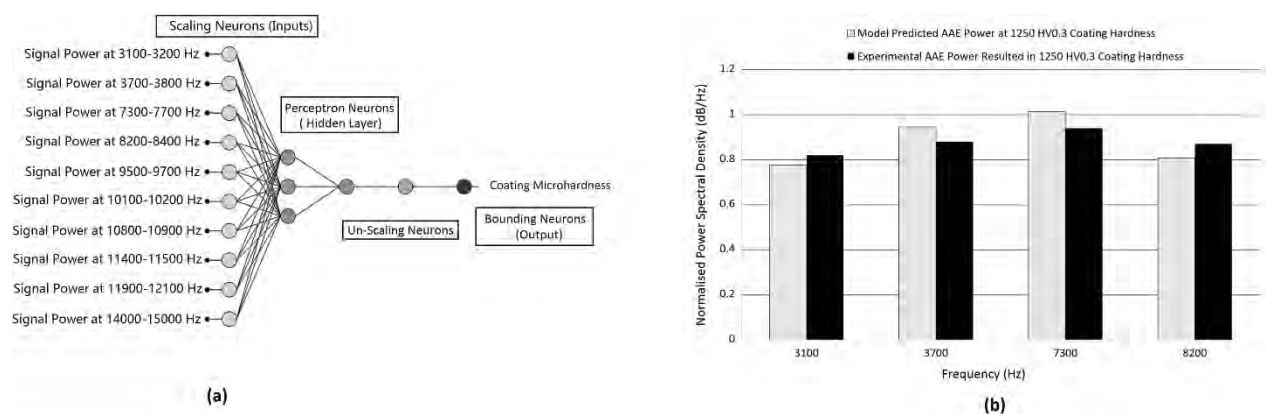


Figure 2.4 a) Neural network architecture used to predict the coating microhardness as a function of frequency power inputs, b) Unseen data predictive capability of the ANN. The spray distance was fixed at 120 mm, the powder feed rate was 2 kg/h, the traverse speed was 600 mm/s, and the impact angle 90° [122]

Despite the good accuracy achieved with simple ANN models, the true power of machine learning can be harnessed by using CNNs alongside sensors. The CNN based models have been used successfully in a variety of applications producing state-of-the-art results. Within the thermal spray research community, the technology uptake has been very slow as opposed to other sectors [123][124][125][126]. A representative example is the work by [127] who developed an intelligent compact, reliable, and cost-effective tool condition monitoring (TCM) system to detect the cutting tool wear in machining of the workpiece material with hardness variation. The generated audible sound signals during the machining process were analysed by support vector machine (SVM) and convolutional neural network (CNN) techniques. The

proposed TCM system shows a high prediction accuracy in detecting the tool wear from the audible sound into the proposed multiclassification wear level in end milling of the nonuniform hardened workpiece. Several other studies are dedicated to advance CNN technologies for interdisciplinary audio classification problems [128][129][130][131].

Audio images are artificially created and used alongside CNNs. Alternatively, directly captured process images can be used to analyse the process characteristics. The application of computer vision artificial intelligence for spray monitoring is an unexplored research area that could offer practical solutions. This approach has been used in several closely related disciplines successfully. In Additive Manufacturing (AM) an automated quality grading system for the AM process using a deep convolutional neural network (CNN) model is presented in [132]. The CNN model is trained offline using the images of the internal and surface defects in the layer-by-layer deposition of materials and tested online by studying the performance of detecting the failure in AM process at different extruder speeds and temperatures. In another study by [133] deep learning methods are used to solve welding problems, such as process monitoring and product quality prediction. An innovative monitoring system is presented in [134] capable of diagnosing the penetration state during the laser welding process. The precision and recall ratios of the CNN found to be higher than those of other methods such as a histogram of oriented gradients and local binary pattern.

In additively manufactured metamaterials the parts exhibit high variability in their mechanical properties than wrought materials, placing more stringent demands on inspection, part quality verification, and product qualification. The study by [135] shows that convolutional neural networks can directly predict the energy required to compressively deform gyroid and octet truss metamaterials using only optical images providing a fast and inexpensive screening tool for predicting properties of 3D printed lattices. Another proposed online monitoring method [136] relies on a deep CNN that takes a real metal AM part's image as input and the part quality

categories as network outputs while a vision system with high-speed camera was used by [137] for process images acquisition. The features of different objects including melt pool, plume and spatter were extracted based on the AM process understanding. This work also demonstrated that CNN modelling is a promising method for quality level identification.

2.3. Computer Vision and Machine Learning for Microstructural Characterization, Analysis and Standardisation

2.3.1 Challenges

Microstructural science has progressed significantly but a principal microstructural metric remained intangible. Instead, each individual metric was developed and used on a case-by-case basis. The rationale is that the visual space of microstructural images is immense, and this ends up in the two fundamental problems of quantitative microstructural science. The primary problem is the identification of the measurable features provided that for any given coating microstructure it is not always self-evident which metric is most strongly associated to the property of interest. In a typical industrial quality department, records of conforming coating images are used, alongside standard metallographic analysis procedures, to compare and assess the microstructure of a coating produced with the same process using the same material composition. Ultimately, deciding which coating microstructure conforms to OEM specifications requires human judgment, knowledge and experience. The outcomes can be uncertain or perhaps irreproducible. For this reason, the author introduces in the following sections several original Machine Learning Computer Vision techniques for the establishment of standard metrics in relation to image similarity and coating quality classification.

The second fundamental issue is the measuring methodology and the equipment that should be used. Some microstructural metrics are relatively straightforward to obtain, others are not. Revealing the true structural and mechanical properties is of cardinal importance for the

optimized use of thermal and cold sprayed coatings. The study by [138] presents several results indicating that new and robust laboratory techniques can reveal the true properties of thermal spray coatings previously completely ignored, causing serious concerns regarding the quality standards developed decades ago. The indicated measurement inaccuracies for testing thermally sprayed coatings suggest that international standards (such as QS9000, ISO17025, AS9003, and ISO10012), should be reviewed and updated taking into account more advanced characterisation techniques. For example, considering a typical appearance of an MCrAlY coating in light optical microscopy it is trivial to distinguish between oxide stringers and delaminations. Other typical issues may ascend from grinding and smearing. The most critical are the filled pores and partially hidden cracks in metallic coatings where segmentation is problematic and usually is left to the subjective judgment of a human expert. The ASTM [139][140] guides cover recommendations for sectioning, cleaning, mounting, grinding, and polishing to reveal the microstructural features of thermal sprayed coatings (TSCs) and the substrates to which they are applied when examined microscopically. Adherence to this guide can provide practitioners with consistent and reproducible results, however more studies are needed that would take into account advanced technologies developed after the last revision of this guide.

2.3.2 Machine Learning Prospects

Structure-processing-property connections rely on accurate microstructural measurements in the way they are made. However, human judgement is required in many microstructural analysis tasks entailing elements of subjectivity that makes automation trivial. To alleviate this fundamental susceptibility of human experts to bias, artificial intelligence methods, including CV and ML, can learn to replicate human visual judgments in an objective way and appear to be good candidates to perform these tasks in an autonomous and efficient manner. In several studies CV and ML approaches have been applied to develop a more general approach to

quantitative microstructural analysis [141]. A comprehensive review has been provided also by [142], however no data were presented for spray coated materials. In [143] a combination of CV and ML has been utilized to classify and to detect alloy microstructures using image process, texture features and neural network. In [144] computer vision and machine learning methods were applied to the challenge of automatic microstructure recognition of dendritic morphologies.

Two classification tasks were completed, and involved distinguishing between micrographs that depict dendritic morphologies from those that do not encompass this particular microstructural feature and from those micrographs identified as depicting dendrites. Support vector machine, voting, nearest neighbours, and random forest models were used to perform this classification task achieving classification accuracies of $97.37 \pm 3.33\%$. The literature reveals [145] that CV and ML methods is a promising direction for creating tools that capture rich and complete microstructural information in a quantitative, objective, and general manner but there is a lack of studies directly addressing the microstructural analysis challenges in spray coated materials.

2.3.3 Coating microstructure similarity index as a conformance and quality metric

In thermal spray shops it is common practise to compare visually coating microstructural images from optical light microscopy methods with conforming coatings provided by equipment manufacturers, powder suppliers and OEMs. Although, coating qualification is a complex process involving several steps, this first visual inspection step plays an important role and dictates the actions required to improve the coating quality through parametric process optimisation. In practise, this is often a truly challenging task because images coming from different sources are subject to distortions during acquisition, processing, compression, storage, transmission and reproduction, any of which can lead to degradation of visual quality and

different contrast and luminance levels. The main objective of image similarity research is the development of quantitative measures that can automatically predict objectively the perceived image structural similarities. The nearest available study of microstructural material characterisation using image similarity approaches is presented by [146] but this study does not focus on spray coated microstructures. These approaches were based on metrics related to natural properties or metallurgical features contained in the microstructural images under investigation. Although only four microstructural images were used as the analysis targets, the different employed methods exhibited consistent evaluation results, indicating their capacities for identifying the image similarity.

The simplest and most widely used full-reference quality metric is the mean squared error (MSE), computed by averaging the squared intensity differences of testing and reference image pixels, along with the related quantity of peak signal-to-noise ratio (PSNR). These are attractive because they are simple to calculate, have clear physical meanings, and are mathematically convenient in the context of optimization. But they are not very well matched to perceived structural coating similarities [147][148][149]. MSE is simple to implement but when using it for similarity tasks becomes problematic. Large distances between pixel intensities do not necessarily mean the contents of the images are dramatically different. The implementation is based on the approach by [150][151] and is driven by the human visual system notion that is highly adapted to extract structural information from the viewing field. The luminance of the cross section of a coating microstructure being observed is the product of the illumination and the reflectance, but the structures of the coating are independent of the illumination. Consequently, to explore the structural information such as porosity, oxides, cracks etc. in an image, we wish to separate the influence of the illumination. The features in the coating microstructure are defined as those attributes that represent the structure of the coating

independent of the average luminance and contrast. The approach by [150] separates the task of similarity measurement into three comparisons: luminance, contrast and structure.

The resulting similarity measure (SSIM index) between two images x, y with mean values and standard deviations $\mu_x, \mu_y, \sigma_x, \sigma_y$ respectively (as implemented in Appendix C) takes the following form:

$$\text{SSIM}(\mathbf{x}, \mathbf{y}) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}. \quad (1)$$

$$\text{SSIM}(\mathbf{x}, \mathbf{y}) = [l(\mathbf{x}, \mathbf{y})]^\alpha \cdot [c(\mathbf{x}, \mathbf{y})]^\beta \cdot [s(\mathbf{x}, \mathbf{y})]^\gamma \quad (2)$$

where:

L : The dynamic range of the pixel intensities (the maximum possible pixel value). For an 8-bit image, $L=255$.

C_1 and C_2 : Constants that are introduced to avoid division by zero for cases where the denominators become very close to zero.

μ_x, μ_y : Mean intensity values of images $\mathbf{x}$ and $\mathbf{y}$, which represent the luminance comparison component of SSIM.

σ_x^2, σ_y^2 : Variance of image $\mathbf{x}$ and image $\mathbf{y}$, representing the contrast comparison.

σ_{xy} : Covariance of $\mathbf{x}$ and $\mathbf{y}$, representing the structural comparison part of the equation.

The SSIM index ranges from -1 to 1 , where a value of 1 indicates perfect similarity.

2.3.4 Microstructure Resolution Enhancement using Machine Learning

Modern material research heavily relies on accurate representation of digital microstructure images that deliver huge amounts of data for visualization and characterization to numerical modelling. Most efforts are focusing on the development of tools and methods that take advantage of digital imaging [152][153][154][155]. Typical approaches include the scanning electron microscopy (SEM), and electron backscatter diffraction (EBSD) techniques while 3-dimensional (3D) visualization and characterization of digital microstructures using EBSD

combined with a focused ion beam (FIB) or Xray tomography [156] have nowadays been extensively used. Similarly in the modeling sector, digital images are used to predict and analyze a wide variety of microstructure-sensitive properties such as strength and ductility, thermal properties, corrosion, recrystallization, phase transformations and wear rates [157].

They key questions if data fidelity is compromised during acquisition or processing has been addressed in a very interesting work by [158] who defined and measured the missing information in digital images. A typical example is the loss of information due to instrumental resolution in EBSD method. Since most material properties are closely linked to geometric details in the microstructure, a compromised image could have a knock-on effect when microstructure-based models are used to study the mechanical behavior of materials and coatings. For instance, the wear behavior of carbide composite coatings with complex microstructures is governed by local phase morphology, meaning that an accurate and precise representation of the local structural features is critical in understanding the damage mechanism. Consequently, restoration and regeneration of information lost during the acquisition or processing of digital images can be enormously valuable for accurate and comprehensive materials analysis specially when combined with modelling and simulation approaches.

The approach that has been used the last decades to address resolution issues is based on low resolution upsampling through interpolation techniques. These techniques include the nearest neighbor, bicubic, or bilinear interpolations. More advanced super resolution techniques include reconstructing high-resolution microstructures from existing high resolution 2D distance correlation functions (DCFs) using several algorithms [159][160][161]. Despite the better performance compared to traditional interpolation methods, these methods require correlation functions and are based on iterative upsampling of low-resolution images. This

approach may be computationally expensive and time-consuming while human intervention, prior knowledge and intuition is required to achieve the best possible outcomes.

In the data-driven era amazing new super resolution techniques have emerged the past decade with those implementing CNNs thriving in multiple sectors [162][163][164][165][166][167][168]. Deep learning mostly for SEM and FIB image upscale has been successfully used by several groups [169][170][171] demonstrating that regeneration of data fidelity within the material context is feasible and very promising. Nevertheless, there are many other digital microstructures that are in demand of SR imaging, such as EBSD image data, which contains more information than SEM images. A very interesting study by [172] addresses the issues of jagged feature edges and oversimplification of features upon resolution loss with the use of ResNet [173] to super-resolve low resolution EBSD image data.

2.4 Improvement in thermal spray processing using Simulation, in process monitoring and post spray processing of data

Given the rapid changes in this field, it is challenging to understand both the breadth of opportunities and the best practices for their use. In this section, the author provides specific examples in which machine learning can have significant impact. The accuracy and domain of applicability of the identified best in class models shown in Table 2.1 are discussed and some opportunities and challenges are highlighted. While the adoption of AI is still in its early days, the examples presented in this section suggest that it is already reaping meaningful rewards.

Description	Approach	References
Coating Optimization	Semi-Analytical Experimental Based Approaches	[26],[31]
Robot Path Optimization	Self-Organizing Maps and Geodesic-Based Model	[41],[25]
Process Monitoring & Control	Artificial Neural Networks Coupled with Sensors	[109],[127],[122]
Microstructural Characterization & Defect Detection	Convolution & Deep Neural Networks	[143],[144],[145]
Conformance	Computer Vision Structural Similarity	[150],[151]
High Fidelity Image Analysis	Image Super-Resolution	[162],[172],[173],[174]

Table 2.1. Literature review summary highlighting the most promising approaches to address the current thermal spray industry challenges.

Chapter 3 Process Monitoring and Control using CNNs

3.1 Methodology Audio and Image

The methodology for "Process Monitoring and Control using CNNs" in the context of thermal and cold spray processes involves several key steps:

Audio analysis

1. **Data Representation and Acquisition:** The methodology begins with deciding on the data representation for audio tasks, choosing between raw audio waveforms or 2-D spectrograms, with the latter being increasingly popular due to compatibility with CNNs. For audio data acquisition, a single microphone with a preamplifier in a thermal spray booth is used. An example involves estimating the powder feed rate during spray from acoustic emissions amidst background industrial noise.
2. **CNN Architecture and Training:** The CNN architecture is tailored to process these audio files or spectrograms. Despite a small number of audio files, the CNN demonstrates an excellent learning curve and ability to predict the powder feed rate in the HVOF jet, as shown by a confusion matrix.
3. **Model Optimization and Application:** The model can be enhanced by increasing the dataset size and developing specific feature vectors for thermal spray. Adjustments to CNN activation functions and architecture are also considered for optimal performance. Practical applications include remote control process troubleshooting and real-time monitoring systems, with the potential to classify various conditions such as gun operating status and nozzle wear.

Image analysis

Extension to Image and Video Data: Beyond audio, the methodology extends to processing real-time images or video streams for monitoring the operating state of a process. CNNs are trained to classify images of the high-velocity jet during spray under different conditions.

1. Data Representation and Acquisition: images of the high-velocity jet during spray under different conditions are acquired by a standard DSLR.
2. CNN Training on Image Data: A CNN is trained to identify features like the presence of powder in the flame, classifying images into categories. The accuracy of the model is high on both training and validation sets, and class activation maps are used to assess the model's focus and generalization ability.
3. Implementation for Holistic Monitoring and Control: With the availability of process-related images, a holistic model can be trained using optimized CNN architectures. This approach includes real-time image processing for evaluating different process parameters, exploring alternative hardware designs, and validating analytical process models. In production, these systems aid in computer-assisted monitoring and real-time process control, with automated adjustments based on process feedback.

In summary, this methodology integrates CNNs for both audio and image processing to monitor and control thermal and cold spray processes, emphasizing the development of models that can accurately classify various process conditions and parameters.

3.2 Results and Discussion

In the literature review several approaches were discussed with emphasis on modern ML techniques including simpler ANNs and more advanced CNNs that have been successfully used in other fields but not to great extent for process monitoring and control of thermal and cold spray processes. In relation to process monitoring using audio files and simple ANNs,

feasibility studies have been published in the field of thermal spray technology. In the field of additive manufacturing, CNNs have been used successfully and appear to be superior to conventional monitoring methods. In the absence of reported studies related to thermal spray, our recommendation for further CNN related research is based on two implementation models that were developed to serve as validation reference studies.

In the context of thermal and cold spray, to build a model for audio tasks the first step is to decide what kind of representation to use for the data. Models could be built using the raw audio waveform or 2-D representation of the audio in the form of spectrograms that have become increasingly popular in recent times because they work well with Convolutional Neural Networks. An example of audio classification during HVOF spray is presented in Appendix A. The selected attribute was the estimation of the powder feed rate during spray from acoustic emissions in the presence of background industrial noise. For the data acquisition in raw audio format, a single microphone with a preamplifier in an industrial thermal spray booth was used.

The microphone Model 378A21 from PCB Piezotronics has a frequency range from 4 Hz to 25 kHz and distortion limit of 160 dB with a noise floor of 19 dBA for data processing. The experiments were performed with an HVOF gun traversing linearly over the substrates using a robotic arm. The spray angle was fixed at 90 degrees, the SODs at 120mm while the feed rate was altered and individually controlled. This allowed for studying the influence of the powder loading within the flame on the generated acoustic signals inside the spray booth. The powder feed rates were varying from 0 kg/h to 2 kg/h.

Class	Value	Number of samples
No Powder	0 kg/h	171
Low Powder	0.2 kg/h	171
Medium Powder	1 kg/h	171
High Powder	2 kg/h	171

Table 3.1 Audio dataset

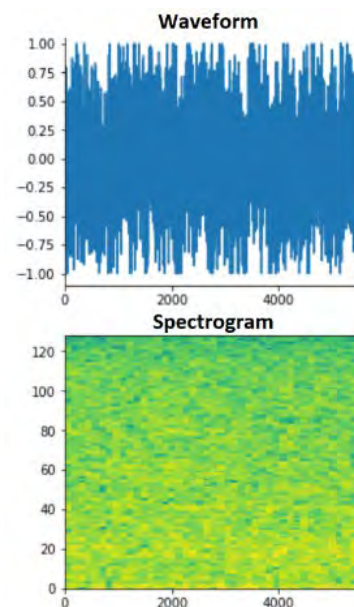
The original dataset consists of 684 WAV audio files. The audio files correspond to four classes of 171 samples corresponding to 0 kg/h , i.e. “No Powder”, 0.2 kg/h “Low Powder”, 1 kg/h "Medium Powder" and 2kg/h "High Powder" (Table 3.1). The dataset was split into training, validation and test sets using a 80:10:10 ratio, respectively. The audio files are initially binary files, which are then converted into a numerical tensor. The audio files are loaded using the *tf.audio.decode_wav* Tensorflow function, which returns the WAV encoded audio as a Tensor and the sample rate. A WAV file contains time series data with a set number of samples per second. Each sample represents the amplitude of the audio signal at that specific time. The sample rate for this dataset is 16kHz. normalized to the range [-1.0, 1.0]. The waveforms are converted into spectrograms represented as 2D images. This is done by using the short-time Fourier transform (STFT) to convert the audio into the time-frequency domain (segment length equal to 255 samples with overlap of 50%, using a standard window function). A Fourier transform (*tf.signal.fft*) converts a signal to its component frequencies, but loses all time information. The STFT (*tf.signal.stft*) splits the signal into windows of time and runs a Fourier transform on each window, preserving some time information, and returning a 2D tensor that standard convolutions can be used with. The waveforms should have the same length, so that when converted to spectrogram images, the results retain similar dimensions. This can be done by zero padding the audio clips that are shorter than one second.

The CNN architecture (Figure 3.5a) and an example of the converted waveform to spectrogram for a random sample in the dataset are shown in Figure 3.5b. Despite the relatively small number of audio files used in this example and the need for further CNN optimisation the model shows an excellent learning curve as shown in Figure 3.6.a and very good ability to predict the powder feed rate in the HVOF jet from raw audio files as shown in the confusion matrix in Figure 3.6.b. The confusion matrix shows the performance of the classification model. For powder feed rates in a particular actual class, the rows tally the predicted class assignments by the classifier. The upper-left to lower-right diagonal entries give the number of accurate classifications for each actual class.

Input shape: (124, 129, 1)
Model: "sequential"

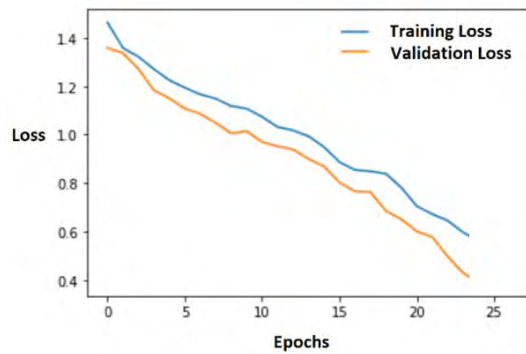
Layer (type)	Output Shape	Param #
resizing (Resizing)	(None, 32, 32, 1)	0
normalization (Normalization)	(None, 32, 32, 1)	3
conv2d (Conv2D)	(None, 30, 30, 32)	320
conv2d_1 (Conv2D)	(None, 28, 28, 64)	18496
max_pooling2d (MaxPooling2D)	(None, 14, 14, 64)	0
dropout (Dropout)	(None, 14, 14, 64)	0
flatten (Flatten)	(None, 12544)	0
dense (Dense)	(None, 128)	1605760
dropout_1 (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 4)	516
Total params: 1,625,095		
Trainable params: 1,625,092		
Non-trainable params: 3		

(a)

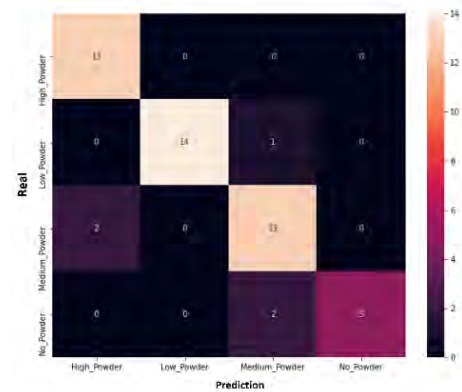


(b)

Figure 3.5. a) CNN architecture, b) Waveform conversion to Spectrogram



(a)



(b)

Figure 3.6 a) Training and Validation Loss b) Confusion matrix

This model can be further improved by increasing the number of samples in the dataset and by developing thermal spray specific feature vectors. Further optimisation can be achieved by altering the CNN activation functions as well as altering the model architecture to achieve the optimum performance. In practise this method can be used for remote control process troubleshooting where the operator records an audio file which is then sent and analysed by quality department personnel or as a real-time process monitoring system. In this simple example only four classes have been used. With sufficient computational resources, the model could be trained to account for the gun operating status, nozzle wear, gas leaks and to predict several key coating properties. There are several challenges requiring more focused research in this field stem from the transducer noise, background noise, various loudness levels, various time durations and finally a large number of process raw acoustic data to feed into the algorithm.

Further to acoustic emission process monitoring, real-time processing of images or video streaming can be used according to the literature to assess the operating state of a process. More details can be found in Appendix B. The key objective of the machine learning approach is to bridge the gap from the material deposition process at the hardware level, to the characteristics of the coating that is produced at the end of the process. To achieve this, CNNs could be trained

to classify images of the high velocity jet during spray under different operating conditions. Even though this is not directly characterizing the resulting coating quality, it provides evidence for the potential of machine learning methods as distinctions are made prior to coating formation

To demonstrate the potential of CNNs, different datasets were gathered and labeled. For any dataset, images of an HVOF torch during operation were captured at different operating conditions such as different gas flow rates and powder feed rates. For each dataset the operating condition was labelled. A database of spray process images was finally produced to train the CNN model. The experimental setup was intended to be simple and minimal. Therefore, a setup including only a standard DSLR camera was used for capturing images of the spray process at different angles and distances.

The images gathered were reduced in size before being used in the CNN. Several input sizes were tested and the minimum image resolution of 50×50 pixel found to have the highest accuracy utilising the minimum computational resources. The approach in the model used in this study is to reduce the complexity of the model as much as possible, without lowering the accuracy of the model. To achieve this, many different architectures were tested and models with different complexity levels were used to train for the spray images, and the results were compared to find the most effective yet light model that could identify the presence of powder in the flame. It was discovered that a simple model with 3 convolutional layers with 3×3 receptive fields, were just as effective as larger and more complex models, and in fact increasing the effective receptive field size past 5×5 had negative effects on the model's ability to reach higher accuracy. Also, the number of convolution filters within each layer were minimized as well and the final result of the experimentation in the model led to an architecture . All layers, except for the final layer have ReLu activation functions, while the last layer has a SoftMax activation function. As evident, there are three convolution layers followed by max

pooling layers after each convolution layer, and after the third max-pooling layer a fully connected layer of size 1024 exists followed by another layer with similar size, with a drop out of density 0.2 placed between the two, and finally the model ends with a SoftMax layer with the size being set to the number of categories at the output, which is determined by the number of experimental operating conditions involved in each dataset. For the training of the model itself, the training algorithm the Adam optimisation technique was used with sparse categorical cross entropy loss function.

A CNN with this structure was trained on the available images to identify the presence of powder in the flame and classify any given images into two categories (Class 1: “Powder”, Class 2: “No Powder”). The most accurate model after the training for this dataset (Figure 3.7) had a 98% accuracy on the training set and was 97% accurate on the validation set, which means that it can be expected that this model would be 97% accurate in classifying any image that falls under the two categories the model was trained for (Figure 3.8.a). A random test input image from the validation set and its classification probabilities are shown in Figure 3.8.b. To further assess the generalisation ability of the algorithm, the class activation maps of an HVOF jet without powder in the flame are plotted in Figure 3.9. The activation map for a given class is regarded as a weighted sum over its feature maps out of the last convolutional layer. A global average pooling layer is used to convert a feature map into a single value, and acts as the glue for calculating the associated weights. From Figure 3.9 it is evident that the model is trained correctly, focusing on the jet area to classify the process. In essence, the convolutional layers extract important features from the image while reducing the size of the image, hence eliminating irrelevant information.

Model: "sequential_2"		
Layer (type)	Output Shape	Param #
conv2d_3 (Conv2D)	(None, 48, 48, 16)	448
activation_5 (Activation)	(None, 48, 48, 16)	0
max_pooling2d_3 (MaxPooling2D)	(None, 24, 24, 16)	0
conv2d_4 (Conv2D)	(None, 22, 22, 32)	4640
activation_6 (Activation)	(None, 22, 22, 32)	0
max_pooling2d_4 (MaxPooling2D)	(None, 11, 11, 32)	0
flatten_2 (Flatten)	(None, 3872)	0
dense_3 (Dense)	(None, 32)	123936
activation_7 (Activation)	(None, 32)	0
dropout_2 (Dropout)	(None, 32)	0
dense_4 (Dense)	(None, 2)	66
activation_8 (Activation)	(None, 2)	0
Total params: 129,090		
Trainable params: 129,090		
Non-trainable params: 0		

Figure 3.7 Model parameters and CNN architecture

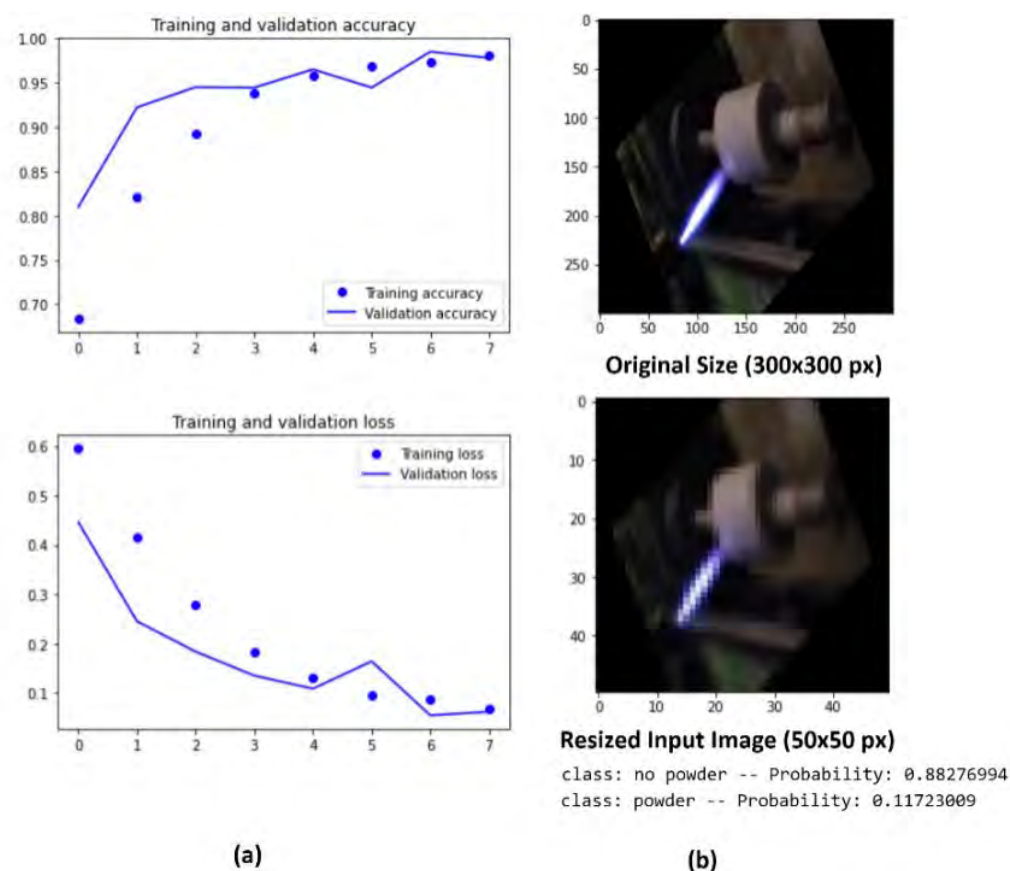


Figure 3.8 a) Training and Validation Accuracy after 7 epochs, b) Original Input image that has been resized and classified as no powder flame with 88% confidence.

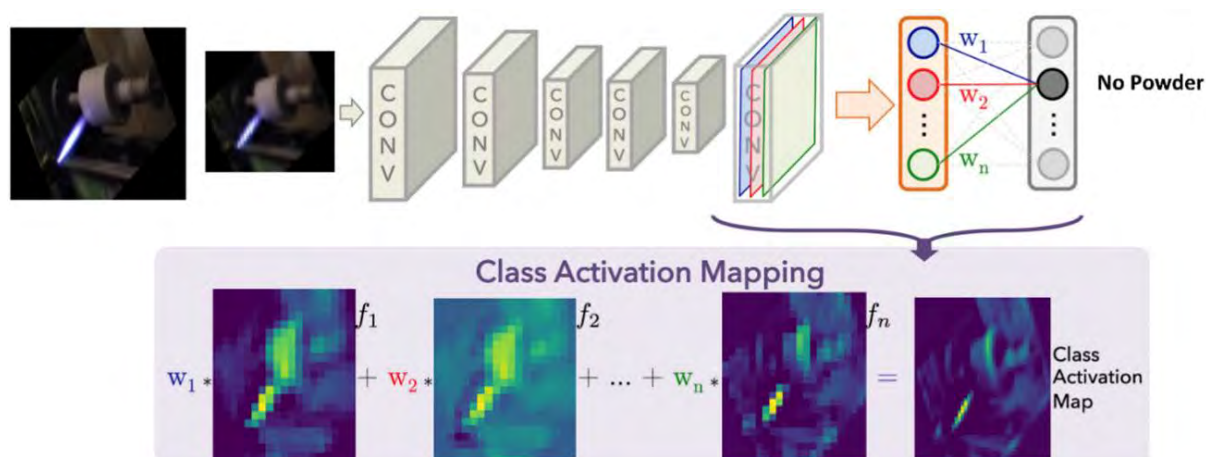


Figure 3.9 Complete activation map process showing the classification focus areas in the jet region of a High Velocity Oxy/Fuel System operating without powder in the flame. The dark areas contribute less to the final image classification, while bright colour shows the model focus areas, namely the flame and the substrate area in Figure 3.9.

Provided that process related images are available, one can use the implementation code provided in Appendix B with optimised CNN architectures to train a multi-parameter holistic model. The addition of a real-time image processing script in the main body of the provided code will allow to use this approach for the evaluation of the effect of different process parameters, explore alternative hardware designs, allow for better record keeping during process experimentation, and for validating analytical process models. During actual production, these systems can be used for computer-assisted monitoring. Different process measures are computed and can be presented to the operator through graphical displays. Automatic alarming is possible by comparing these computed measures to expected values and tolerances. Ultimately, the method can be effective for real-time process control where the process parameters are automatically adjusted on the basis of the process feedback.

Chapter 4. Process Monitoring and Control using YOLO

4.1 Methodology

The methodology for "Process Monitoring and Control using YOLO" in the context of thermal spray process involves several critical components:

1. **Combining Acoustic Emissions and Visual Status:** This approach integrates acoustic emissions with the visual observation of the torch for comprehensive monitoring. It's particularly effective for identifying issues like nozzle wear, clogging, gas leaks, and other disturbances that can affect coating quality.
2. **YOLO Architecture for Prediction:** The YOLO (You Only Look Once) architecture is employed to monitor the spray process using High Velocity Oxy-Fuel systems. This involves analyzing various visual features of the torch's operation, such as the jet's shape and flame color intensity.
3. **Region of Interest (ROI) Identification and Classification:** Specific regions within the frame are manually identified as ROIs, (or anchor boxes) containing information about particle temperature, velocity, and coating sound features. The YOLO architecture automatically extracts features from these ROIs and classifies coating parameters like air flow, jet temperature, pressure, and resulting microhardness.
4. **Dataset Development:** For object detection and classification, a multi-class labeled dataset is created, based on experimental observations or numerical modeling data. This includes monitoring flame temperature, velocity, and airflow, along with acoustic emission spectrogram analysis. The experimental data were provided by Castolin Eutectic.
5. **Torch Fluid Dynamic Simulations:** 3D simulations of the spray torch under various gas flows are conducted to understand where to look for information in images. These

simulations help in identifying visible manifestations of the jet's physics. The CFD work has been conducted at Castolin Eutectic using the commercial CFD solver Ansys Fluent.

6. **Acoustic Emission Classification:** Acoustic emissions, particularly the jet noise and its variations, are analyzed. This involves studying the impact of powder in the flame on velocity profiles and the relationship between noise generated and coating microhardness.
7. **Object Detection & Classification Using YOLOv5:** The YOLOv5 architecture is used for real-time object detection. This includes pre-processing steps like data labelling and anchor box placement, image acquisition, and training. Different versions of YOLOv5 models are benchmarked to find the best balance between speed and accuracy.
8. **Model Training and Evaluation:** The model training involves data augmentation, loss function analysis, and performance metric evaluation. Techniques to combat overfitting, such as regularization and early stopping, are employed.
9. **Industrial Implementation and Limitations:** The model is adapted for use with specific spray torch characteristics and can be converted for mobile device compatibility. Limitations include the need for new image capture and annotations for different classes of interest.
10. **Manual Real-time Monitoring and Adjustment:** The operator can observe the torch's operating status in real time, with the ability to identify deviations from optimal conditions and make necessary adjustments.

In conclusion, this methodology presents a sophisticated approach to monitoring thermal spray processes by integrating visual and acoustic data using the YOLOv5 architecture, offering significant potential for improved process control and coating quality assurance.

4.2 Results & Discussions

The proposed methodology in this section combines the process acoustic emissions and the visual status of the torch to perform classification of the condition of the spray process. The use of acoustic emissions and visual status of the torch allows for a more comprehensive understanding of the thermal spray process, as it takes into account both the physical and acoustic properties of the process. This approach is particularly useful for detecting and monitoring potential issues in the thermal spray process, such as nozzle wear, nozzle clogging, gas leaks, gun spitting, and other disturbances that can affect the quality of the coating.

4.2.1 Proposed Implementation Strategy and contributions of this work

In this study, we assess the efficacy of the YOLO architecture [175] as a prediction model to monitor the spray process using a High Velocity Oxy-Fuel system. Figure 4.1 displays the torch in action. Oxygen and fuel are combined in a combustion chamber, and the resulting hot gases are accelerated through a supersonic nozzle. The coating material, in the form of powder, is introduced into the jet stream and is propelled at high speed and temperature towards the surface to be coated. The jet's shape, the flame's color intensity, the structure and quantity of shock diamonds are specific to the spray conditions, since they possess distinct visual content and identity that indicate the torch's operating status. This operating status, in turn, governs the coating quality, which is of paramount importance in industrial applications. In Figure 4.1, a typical torch frame is shown and four (4) regions of interest (ROIs) are highlighted, that contain information about particle temperature and velocity and coating sound features (the embedded dynamic spectrogram of the recorded sound at the upper right corner). The features of these ROIs are automatically extracted by the YOLO, which classifies the following coating parameters: a) total air flow, b) maximum jet temperature and c) pressure, as well as d) the resulting microhardness in one of the two possible values. Process condition regions showing the detection anchor boxes and focus areas. Each spray parameter corresponds to a set of

classes. The physics of the coating process allows us to reduce the possible combinations of the aforementioned parameters into two sets of operating parameters, as shown in Table 4.1.

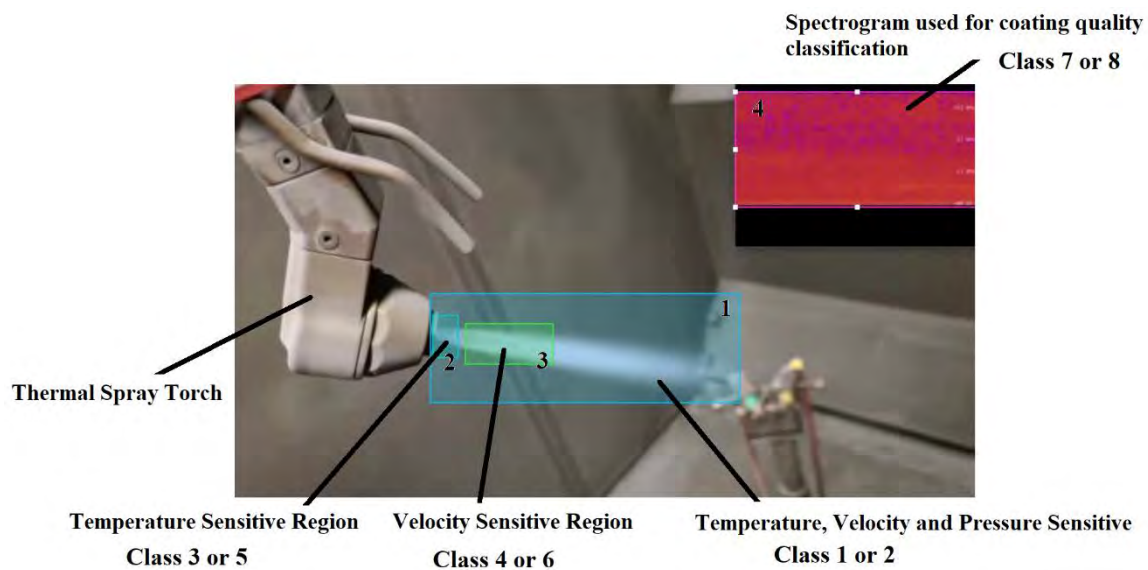


Figure 4.1. A typical frame of an operating torch and class annotations. The four (4) ROIs are highlighted and the classification of the corresponding spray parameter is also shown.

A prerequisite for object detection and classification using CNNs is identifying where to look in an image to assign properties, parameters, or other spray characteristics, which are then used to create a multi-class labeled dataset. These input values must be known and correlated beforehand, either through experimental observations or numerical modeling data. In this research, we have chosen to monitor the flame temperature, velocity, and airflow in conjunction with the acoustic emission spectrogram, which contains information about the combined influence of the spray parameters that have already been correlated with the applied coating quality in our previous study [122].

Figure 4.2 illustrates a flowchart delineating the intricate relationship between various operational parameters of a thermal spray process and their collective impact on the microhardness of the resulting coating. At the core, flame temperature, jet velocity, and acoustic emissions are identified as the primary influencers. Flame temperature, which can be altered by factors such as worn gun parts, incorrect fuel-to-air ratio (stoichiometry), water leaks,

compromised fuel quality, a drop in air pressure, or dehumidifier failure, plays a crucial role in determining the coating's hardness. Jet velocity is equally critical, with its determinants including the condition of the nozzle (worn or clogged), the design of the nozzle, systemic pressure drops, and variations in the total gas flow. Additionally, acoustic emissions during the spraying process—affected by the stand-off distance, jet velocity, and powder feed rate fluctuations—offer indirect cues to the process stability and, by extension, the quality of the coating. The flowchart also suggests a link between flame temperature and secondary factors like air flow, clogging, particle loading, and the resultant jet shape, all converging to influence the final characteristic of interest: the coating's microhardness. The ultimate objective is to sustain a high coating quality level throughout the process in terms of micro-hardness. Micro-hardness has been chosen as the key indicator because it is a property that is immediately affected even by minor deviations from the optimal spray conditions.

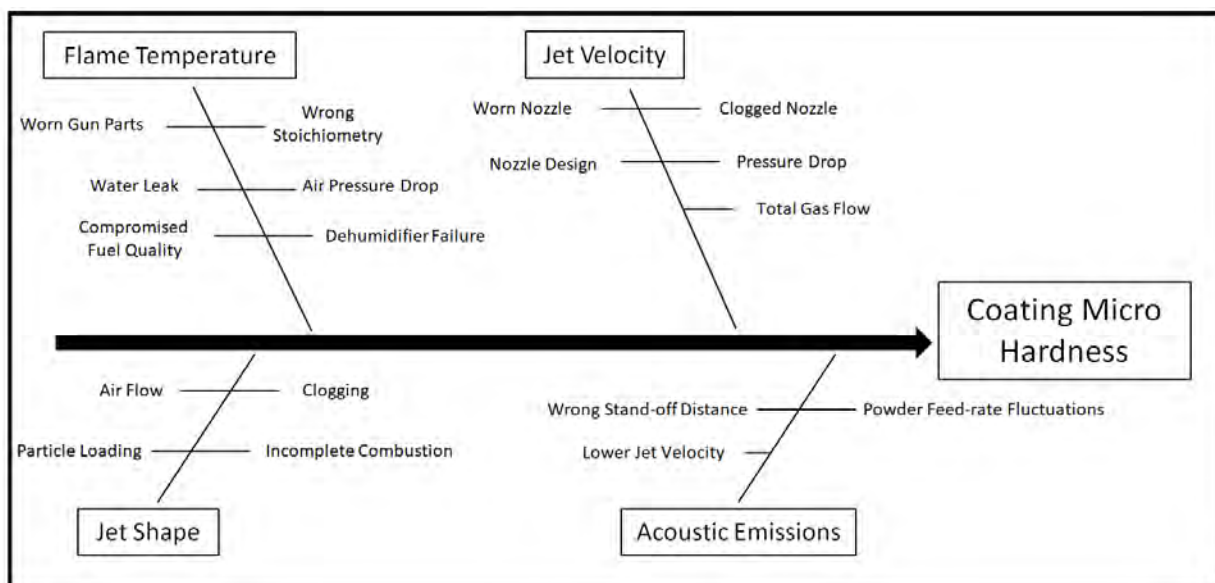


Figure 4.2 Relationship between various operational parameters of a thermal spray process and their collective impact on the microhardness of the resulting coating.

4.2.2 Torch Fluid Dynamic Simulations

In this study, a complete 3D simulation of the spray torch was conducted under various gas flows, as illustrated in Table 4.1, using ANSYS Fluent 16 Academic Edition [18], since the available sensors were unable to measure jet velocity and temperature. The goal was to predict a vector of 2 components, describing the state of the torch, namely jet velocity and temperature (1800K or 2600K and 2000ms^{-1} or 2100ms^{-1}). The simulated process parameters were then employed to capture real process images from the operational torch under the same input conditions. For the scope of this research, only two spray conditions were examined, with each model prediction class displayed in parentheses in Table 4.1. The detailed model implementation can be found in our previous work [122], and only the primary solver settings are summarized in Table 4.2.

Simulating the physics of the process can help to understand where to look for information that can in turn help in the classification problem. The simulated data was primarily utilized to delve deeper into the inherent physics of the process and not for training purposes. Additionally, it served as a directional aid, drawing our attention to specific regions where the physics manifests visibly within the images. The parameters were chosen strategically to anticipate significant alterations in the jet structure. Our intention is to demonstrate the model's capability in highlighting the intricacies of the jet's supersonic structure. Furthermore, this work aims to showcase the model's ability in discerning variations in the shock diamond structure and temperature fluctuations within the flame.

Both experimentally and numerically, it has been observed that a hotter flame results in a brighter jet appearance near the nozzle exit, as indicated by the red boxes in Figure 4.3. When the airflow in the flame increases from 100 SLPM (Standard liters per minute) to 300 SLPM, the temperature decreases by almost 700K compared to the 100 SLPM airflow case. This is

due to the fuel-lean conditions that develop within the combustion chamber. As a result, classes 3 & 5, as displayed in Table 4.1, are detected within the red anchor box regions in Figure 4.3. As the velocity of the supersonic jet increases, its shape transitions from round or circular to a more elongated and narrow structure. The main characteristics that indicate velocity variation are the presence of shock diamonds, their shape, and the total number of visible crisscrossed shock waves. The gas within the jet expands and cools down as it flows through these rarefaction fans and compresses as it passes through the shock diamonds.

In reality, the jet structure does not have sharp, stable boundaries, but rather turbulent ones where jet and ambient gases mix. The mixing layer, which grows, penetrates the supersonic core of the jet. When the mixing layer reaches the jet's axis, the flow becomes subsonic and fully turbulent. Both experimental and modeling images clearly show that the intensity and number of visible shock diamonds increase with velocity. Therefore, the CNN model classification concerning velocity is concentrated on the diamond jet core region, as illustrated by the black anchor boxes in Figure 4.3.

Operating Parameters	A	Simulation Results	Experimental
		Chamber Pressure 0.64 MPa Max Jet Temperature: 2000K (Class 3) (Box Number 2 in Fig.1) Max Jet Speed: 1800m/s (Class 4) (Box Number 3 in Fig.1)	WC-Co (83/17) Coating Microhardness 1000HV0.3kg (Class 7) (Box Number 4 in Fig.1)
Operating Parameters	B	Simulation Results	Experimental
		Chamber Pressure 0.6 MPa Max Jet Temperature: 2600K (Class 5) (Box Number 2 in Fig.1) Max Jet Speed: 2100m/s (Class 6) (Box Number 3 in Fig.1)	WC-Co (83/17) Coating Microhardness 1300HV0.3kg (Class 8) (Box Number 4 in Fig.1)

Class Description in relation to spray parameters.

Table 4.1 Class Description in Relation to Spray Parameters

Ansys® Academic Research	3D/Transient/Coupled Compressible Ideal Gas	Momentum: 2 nd Order Density: 2 nd Order Turbulence κ - ϵ 2 nd Order	Ref (Ref 18)
Turbulence	Realisable κ - ϵ	Enhanced Wall Functions $y^+ \sim 15$ and 10 cells in the boundary layer	Ref (Ref 19)
Combustion	Premixed Species/ Transport	Eddy-Dissipation	Ref (Ref 20)

Table 4.2 Solver Setting and Model

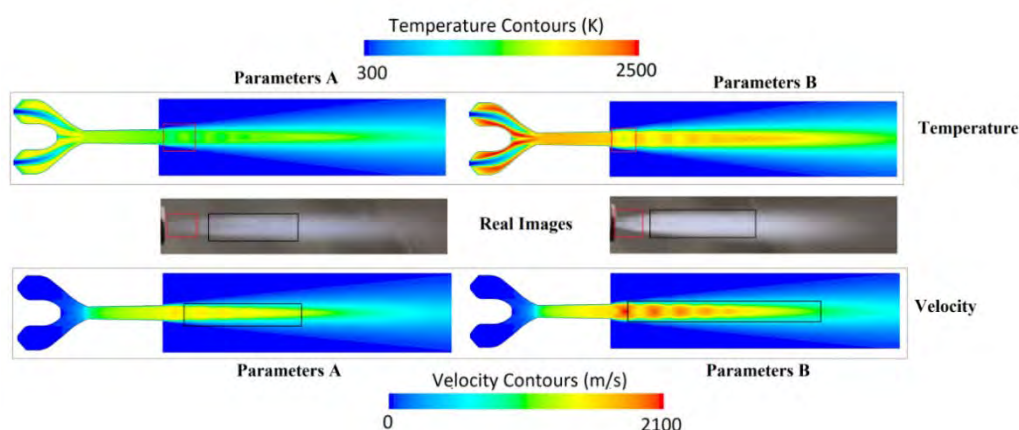


Figure 4.3. Simulated and Experimental images of the HVOF Jet structure showing the location of anchor boxes. The red anchor box classifies the flame temperature, the black anchor box classifies the jet velocity.

4.2.3 Classification based on Acoustic Emissions

The supersonic jet lacks sharp edges, leading to the formation, propagation, and dissipation of eddies, which ultimately results in jet noise. When powder is introduced into the flame, the mean velocity profile is affected, as demonstrated in [122]. Other studies [176][177] have also shown the effect of supersonic jet diameter on aeroacoustics. Typically, a wider jet causes a shift in frequencies where energy peaks occur, and an overall frequency shift of the noise peaks should be expected due to the particle-laden flow in the jet.

Regarding the stand-off distance, the power peaks of a supersonic jet were observed to shift to a lower frequency range when the jet hits a substrate at short standoff distances [122]. At a short distance from the nozzle tip, the jet undergoes rapid deceleration upon impact, creating

large circulation zones (vortex-induced zone) near the substrate. These large turbulent structures give rise to the low-frequency power spectrum. However, when the substrate is moved to a greater distance, the jet experiences more canonical decay, where mixing layer structures at the mid-frequency range dominate the noise generation process.

Previous studies [122] have revealed a strong correlation between the noise generated and the resulting coating microhardness. The most critical frequency range was found to be between 3100 and 10000 Hz. This is the low-frequency range mainly associated with large eddies and the impact of the jet on the substrate. In the absence of a substrate during the process startup stage, higher frequency ranges contribute more to the predicted coating microhardness. Since there is a strong combined influence across a wide range of frequencies, monitoring the entire frequency domain simultaneously is necessary for more accurate predictions. In this study, the frequency domain is analyzed in real-time using the YOLOv5 CNN architecture. This is accomplished by converting the audio file into a spectrogram that is continuously classified alongside the images from the spray gun during operation, as shown in Figure 4.1. The acoustic data acquisition setup and the discovered frequency correlations are described in detail in [122]. A step-by-step process description for converting the acoustic emissions to spectrograms is provided in [178]. Essentially, the velocity of the jet is linked to peaks in energy at high frequencies. The greater the energy peaks at these high frequencies, the faster the jet velocity, and consequently, the impact velocity of the particles. As the jet velocity increases, it is anticipated that the microhardness of the coating will also increase.

The upper right portion of Figure 4.1 showcases a spectrogram, which visually represents sound frequencies on its vertical axis, while the horizontal axis depicts time progression. The varying intensities of red and blue dots within this spectrogram convey the amplitude or loudness of each corresponding frequency at any given moment. By leveraging the capabilities of the YOLOv5 CNN architecture, this model has been trained to interpret these spectrograms

in real-time. This enables the model to discern and classify specific patterns within the spectrogram tied to distinct sound frequencies and their magnitudes. It's noteworthy that coatings with high hardness tend to exhibit certain patterns in the spectrogram that are distinctively different from those of coatings with lower hardness. There are differences in the concentration and distribution of the blue specks between the two classes. The spectrogram that corresponds to lower coating hardness has blue specks that are slightly more dispersed and scattered throughout, especially in the mid-frequency range (around 9.7 KHz). The spectrogram that corresponds to higher coating hardness appears to have a denser concentration of blue specks in the same mid-frequency range.

In Appendix A, the script is designed to capture video and audio in real-time from a camera and microphone, compute the spectrogram of the audio, and overlay the spectrogram on the video frame in the upper right corner. It starts by setting up the video capture from the camera using OpenCV and the audio stream from the microphone using PyAudio. Then, it enters a loop where it continuously captures video frames and audio samples. For each iteration of the loop, the script captures a chunk of audio data and computes its spectrogram using the `scipy.signal.spectrogram` function, which is a 2D representation of the audio signal that shows how the frequencies present in the signal vary over time. The spectrogram is then visualized using the `matplotlib.pyplot.pcolormesh` function, which creates a plot of the spectrogram matrix. The plot created by matplotlib is then converted to an image and resized to one fourth of the video frame size. This resized image is then overlaid on the video frame in the upper right corner. The video frame with the overlaid spectrogram is then displayed in a window using `cv2.imshow`.

4.2.4 Object Detection & Classification

YOLOv5 (You Only Look Once version 5) [175] is a cutting-edge real-time object detection architecture for computer vision tasks. It is designed for speed and accuracy, capable of detecting and classifying objects in real-time even in high-resolution images and videos. The architecture is based on a single convolutional neural network (CNN) that processes the entire image in one pass. The network consists of three main parts: the backbone network, the neck network, and the head network. The backbone network extracts features from the input image using a series of convolutional and pooling layers. The neck network connects the backbone to the head network and is responsible for feature fusion and enhancement. The head network carries out the final object detection and classification tasks using a series of convolutional and fully connected layers. YOLOv5 architecture also employs various techniques such as multi-scale feature pyramid network, cross-scale feature aggregation, and anchor-free detection to enhance overall performance [175].

Before implementing YOLOv5, several pre-processing steps should be taken. One of the first steps is to gather images of the thermal spray process using the two sets of spray parameters shown in Table 4.1. These images serve as the training data for the model. Next, the data needs to be labeled and organized into classes, with each class representing a specific spray parameter. In this case, there are 8 classes corresponding to two spray parameters, with a total of 400 original images per spray parameter, encompassing 80 augmented images. The total number of images is 960. The optimal image size was found to be 640px, trained in batches of 32 per epoch. The next step is to split the data into training and test sets. The training set is used to train the model, while the test set is used to evaluate the model's performance. Generally, it is recommended to use 80% of the data for training and 20% for testing. Next, data augmentation is applied to the training data to increase the dataset size and help the model generalize better. The enhancement process includes rotating the image, modifying the

contrast, and adding artificial noise that might be present in real-world data. This approach helps the model to become unaffected by the lighting conditions in the booth and the specifications of the camera. The training was performed using an Nvidia RTX A5000 graphics card with 24Gb of RAM. It is crucial to note that the dataset comprises images taken at various distances and angles to ensure the trained model can detect and predict all classes under different viewing conditions. This allows the camera to remain fixed in the booth while the torch moves.

In this study, five YOLOv5 models are benchmarked. YOLOv5n is a newly introduced nano model, which is the smallest in the family, intended for edge computing, IoT devices, and OpenCV DNN support. It is less than 2.5 MB in INT8 format and around 4 MB in FP32 format, making it ideal for mobile solutions. YOLOv5s is the small model in the family with about 7.2 million parameters, suitable for running inference on the CPU. YOLOv5m is a medium-sized model with 21.2 million parameters, intended for various datasets and training since it provides a good balance between speed and accuracy. YOLOv5l is the large model of the YOLOv5 family with 46.5 million parameters, ideal for datasets where detecting smaller objects is necessary. Lastly, YOLOv5x is the largest among the five models and has the highest mAP among them. Although it is slower compared to the others, it has 86.7 million parameters.

Increasing the number of parameters in the YOLO model can have both positive and negative effects. On the positive side, more parameters can lead to better performance, as the model has a greater capacity to learn from the data, and can learn more complex features and relationships. This can result in more accurate object detection and classification. However, there are also several drawbacks to increasing the number of parameters. It increases the computational and memory requirements of the model, making it slower to train and unsuitable for devices with limited resources. It also increases the risk of overfitting, particularly if the amount of training data is limited, and results in longer training times. Overfitting occurs when the model learns

the noise in the training data instead of the underlying patterns, leading to poor generalization to new, unseen data.

To combat overfitting, several strategies are adopted in this work. Observing training vs. validation plots helps detect overfitting, especially when training loss decreases, but validation loss increases. Regularization techniques, such as weight decay and dropout, can prevent the model from over-adapting to training data. Data augmentation, which includes applying random transformations to training data, enhances model robustness. For smaller datasets, reducing model complexity or utilizing a simpler version of models like YOLO nano can be beneficial. Early stopping, based on validation loss monitoring, can prevent noise learning, while increasing the training data volume or using transfer learning with pre-trained weights can aid in model generalization.

4.2.5 Discussions and results

To test the model's performance, we acquired 460 frames with a mobile camera inside the booth, during coating with a moving torch and camera. The capture framerate was 60 frames per second and the camera used image stabilization, 1080p analysis, auto iso and exposure settings. The capture was continuous and the camera position was not fixed. Also, the parameter switch did not take place during capture since the system needs several seconds to adapt to new parameter settings. During the first 260 frames the spray parameters were kept constant as in group A and then changed to the values corresponding to parameters B for 200 frames, using a digital mass flow controller. To account for all case scenarios the distance and recording angles were continually altered.

The PR-curve model performance metric is employed to assess the performance of the object detector by measuring the precision-recall tradeoff for varying detection thresholds. A PR-curve plots precision (the proportion of true positive detections among all positive detections)

against recall (the proportion of true positive detections among all actual objects in the image) as the detection threshold changes. The area under the PR-curve, referred to as the average precision (AP), is a standard metric for evaluating the overall performance of the object detector.

A higher AP value indicates that the detector can maintain high precision while also achieving high recall, meaning it can accurately detect a large proportion of objects in the image. In the context of detecting process parameters in thermal spray, a high AP value would suggest that the model can accurately detect a high proportion of the process parameters while maintaining a low false positive rate. A comparison between the smallest YOLOv5n and the largest YOLOv5x models is presented in Figure 4.4. It is evident that the smallest model performs exceptionally well in this particular metric.

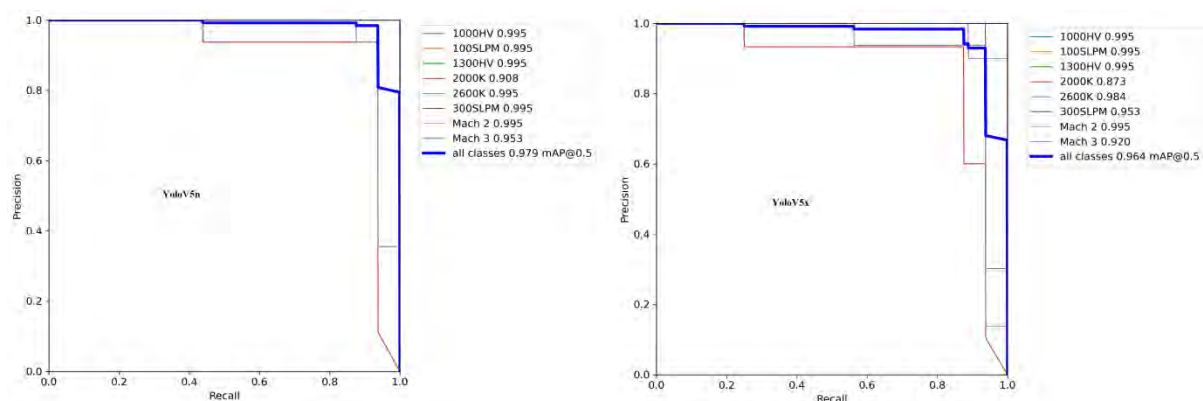


Figure 4.4 PR-Curve comparisons between the YOLOv5n and YOLOv5x models

The R-curve (recall versus confidence threshold) is utilized to assess the performance of a model in detecting objects of interest within an image. Recall is the proportion of true positive detections (correctly identified objects -anchor boxes) among all the actual objects present in the image. The confidence threshold is a value used to filter out detections that the model has low confidence in, thereby enhancing overall detection accuracy.

To interpret an R-curve, one should examine the relationship between recall and the confidence threshold. A higher recall value indicates that the model is accurately detecting more objects, while a higher confidence threshold implies that the model only considers detections it is more confident about. Ideally, a high recall value should be maintained while still preserving a high confidence threshold, striking a balance between detecting all objects and minimizing false positives. In practice, the optimal trade-off point between recall and confidence threshold is determined by the specific use case and the desired precision-recall balance. In this particular case, the YOLOv5x model achieves an average recall of 96% at an 80% confidence level. Similarly, the smallest of the five models, YOLOv5n, performs equally well, attaining a recall of 94% at an 80% confidence level.

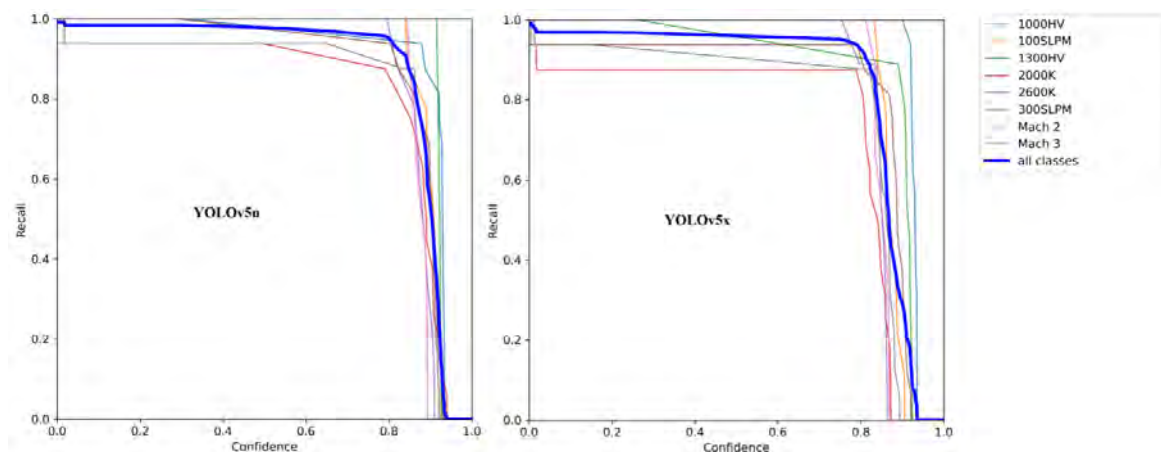


Figure 4.5 R-curve comparisons between the YOLOv5n and YOLOv5x models

Interpreting an F1-curve (Figure 4.6) helps measure the trade-off between precision and recall. The F1-score is the harmonic mean of precision and recall, ranging from 0 to 1, with 1 being the best value. To interpret an F1-curve, one can examine how the F1-score changes with different thresholds for the model's prediction confidence scores. A higher F1-score indicates that the model is producing fewer false positives and false negatives, accurately detecting more relevant objects while minimizing the number of irrelevant objects detected. The F1-curve can

be used to find the optimal threshold for the model's predictions, balancing the trade-off between precision and recall.

For example, all classes in the YOLOv5n model show a precision above 90% at a confidence level above 65% in figure 4.6. This means that the model is correctly identifying objects of each class 90% of the time when making predictions with a confidence of at least 65%. The large model YOLOv5x achieves similar performance but with significantly higher confidence at 75%. However, this doesn't take into account the number of false positives or false negatives the model is producing. To fully evaluate the performance of the model, it's crucial to also examine the confusion matrix in figure 4.7, which considers the number of true positives and true negatives.

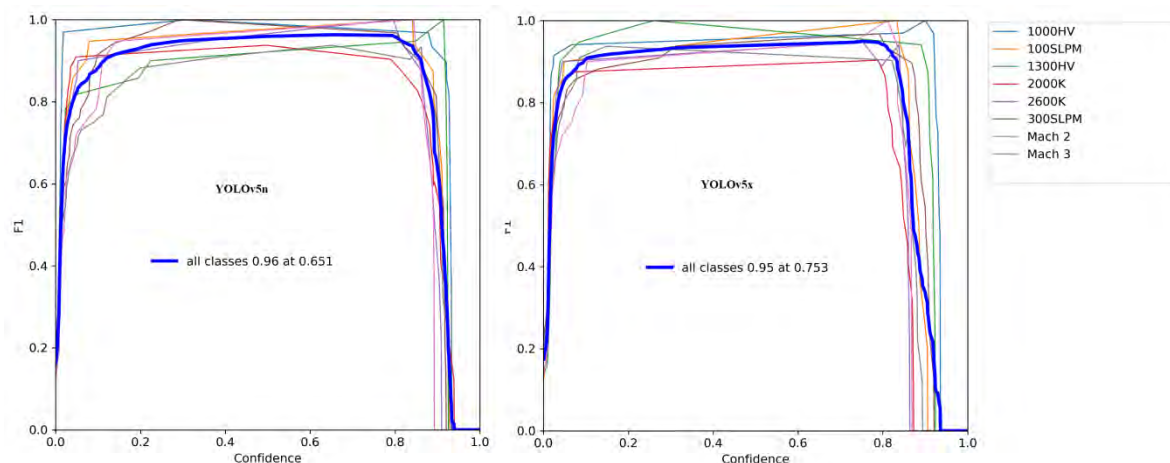


Figure 4.6 F1-curve comparisons between the YOLOv5n and YOLOv5x models

In thermal spray process monitoring obtaining an indication—even with lower confidence—is of paramount importance and emphasizing on recall becomes essential. Based on the provided graphs, the YOLOv5x model consistently demonstrates higher F1 scores across varying confidence thresholds, suggesting a balanced performance in both precision and recall. Moreover, when examining the Precision vs. Recall graphs, YOLOv5x exhibits a curve that's closer to the top right corner, indicating superior simultaneous precision and recall. Therefore,

considering the necessity of high recall and the observed performance in the graphs, YOLOv5x is the recommended model for this particular application.

The confusion matrix is a table used to evaluate the performance of a classification algorithm. It compares the predicted classes (model outputs) with the true classes (target values) in a dataset. The confusion matrix typically contains four types of information: true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN). In the context of YOLOv5, true positives are the number of objects correctly detected by the model, false positives are the number of objects incorrectly detected when no object is present, true negatives are the number of non-objects correctly not detected by the model, and false negatives are the number of objects the model fails to detect. Analyzing the confusion matrix values provides insight into the model's performance and highlights areas for improvement.

Each row of the matrix represents instances in an actual class, while each column represents predicted instances. The cells of the matrix show the number of instances predicted to belong to a particular class when they actually belong to a different class. Ideally, the matrix's diagonal elements should have high values, indicating accurate class label prediction. The off-diagonal elements represent misclassifications, which should be minimized to enhance classifier performance.

Both models perform well in this classification problem. As expected, the YOLOv5n model shows an increased number of background FNs (False Negatives) where the model predicted the background class (i.e., not placing an anchor box at the correct image position) when an object was actually present, meaning the model failed to detect the object in some instances. Similarly, FPs indicate placing an anchor box at a position with no object present. This indicates a lower recall for the object class and suggests the model may need more training data and accurate annotation to match the YOLOv5x model's performance. The model's performance

significantly improves when more parameters are introduced to the architecture, as seen in YOLOv5x.

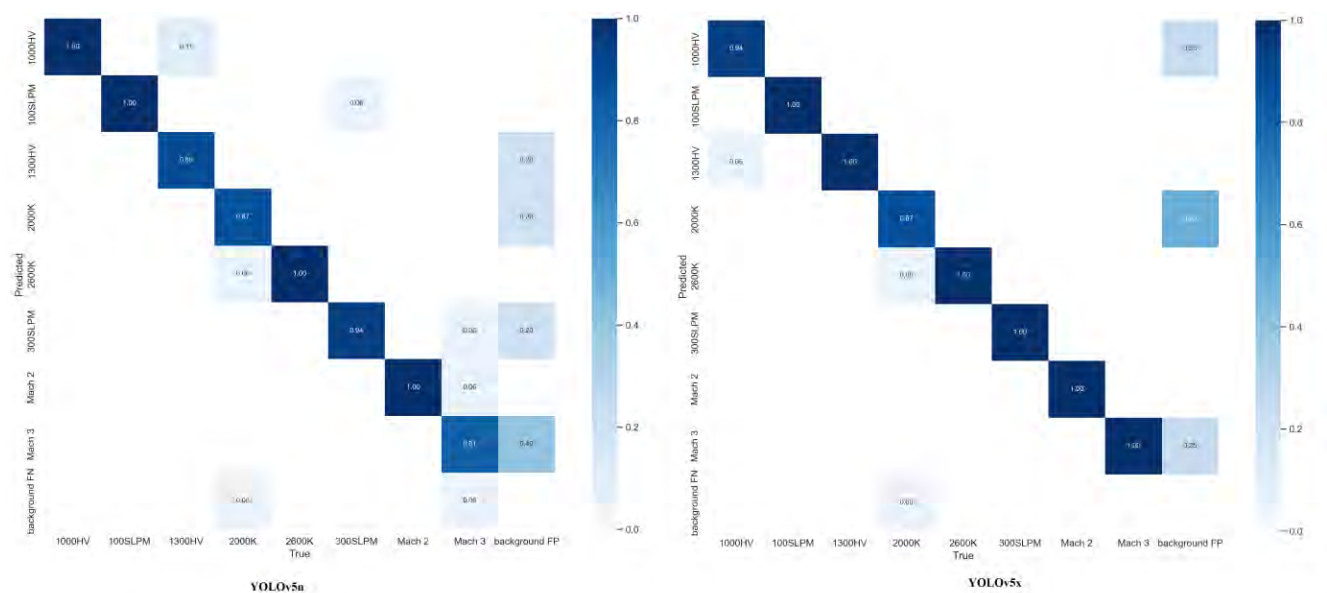


Figure 4.7 Confusion matrix comparisons between the a) YOLOv5n and b) YOLOv5x models

Figure 4.8 illustrates the training progression of the YOLOv5n model over 500 epochs. On the x-axis, we see the number of training epochs, while the y-axis presents various loss functions and performance metrics.

Starting with loss metrics, the Train/Box Loss and Val/Box Loss pertain to the bounding box predictions of the model. These values quantify the discrepancies between the predicted bounding boxes and the actual ground truth, generally computed using the Mean Squared Error / Distance (MSE) of the identified box from the correct one. The differentiation between 'train' and 'val' represents calculations made on the training and validation datasets, respectively. In tandem, Train/Obj Loss and Val/Obj Loss revolve around the model's predicted object score, reflecting its confidence in the presence of an object within the predicted bounding box. A minimized object loss implies accurate object presence predictions. Finally, Train/Cls Loss and Val/Cls Loss focus on class predictions, evaluating the variance between predicted class probabilities and true class labels, via Cross-Entropy loss.

Delving into performance metrics, Object Loss gauges the model's proficiency in precisely identifying both the location and class of objects in an image. Over the course of training, an ideal trend would manifest as a decline in object loss, signaling the model's enhanced prediction accuracy. The mAP (mean Average Precision) serves as a benchmark for evaluating object detection models. This metric computes the average precision across manifold object categories and multiple object instances within each picture. An advanced variant, mAP inrange [0.5:0.95], encompasses calculations at varying IoU (Intersection over Union) thresholds, ranging from 0.5, 0.55, and so on, up to 0.95, in order to generate the PR-curves. IoU assesses the congruence between predicted and actual anchor box overlaps, hence mAP_{0.5:0.95} offers a broader evaluation spectrum of the model's accuracy. Notably, an mAP_{0.5} score surpassing 0.9 is deemed excellent, while anything above 0.6 remains commendable.

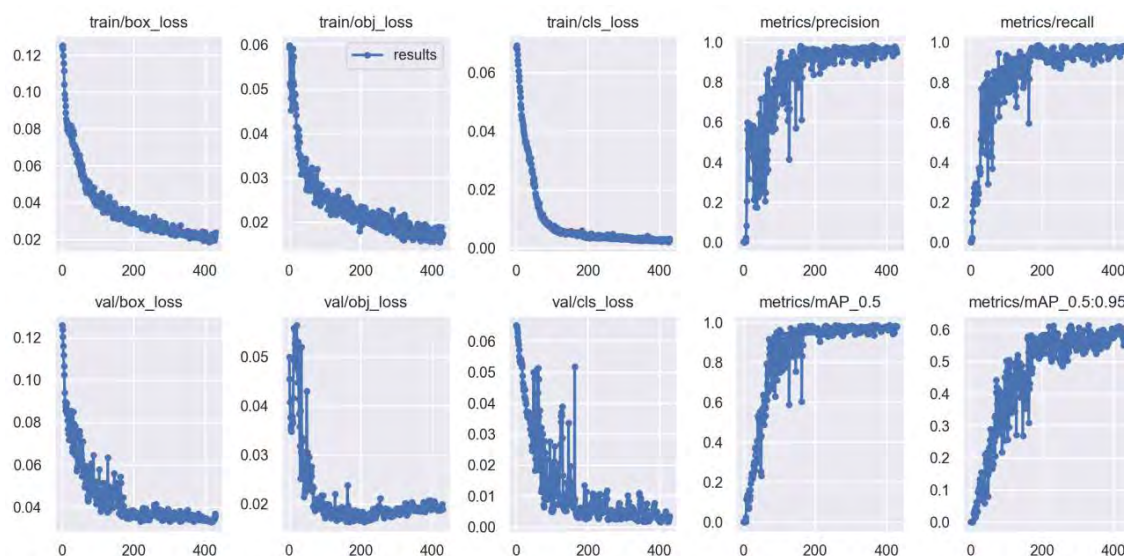


Figure 4.8 Training evolution of the YOLOv5n model over the course of 500 epochs

In Figure 4.8 both the training and validation box losses decrease consistently as the number of epochs increases, converging towards a minimal value. This indicates that the model is progressively learning to predict bounding box coordinates better over time. The validation loss is in close agreement with the training loss, suggesting minimal overfitting in this regard.

The training object loss showcases a clear and steady decline, while the validation object loss exhibits an initial sharp decrease, followed by a period of slight volatility. This oscillation in the validation object loss might suggest that while the model is becoming more confident in predicting the presence of objects in the training data, it occasionally struggles with new, unseen data.

The class loss for both training and validation datasets plummets rapidly in the initial epochs and then remains relatively stable. This indicates that the model quickly learned to distinguish between different object classes and maintained this distinction effectively. Precision starts off at a higher value and slightly oscillates, but it generally remains high. High precision suggests that of all the objects the model predicts, a high percentage are indeed correct. Recall starts at a lower point and sees a steady increase, meaning that the model is gradually getting better at detecting all possible objects in the images. However, there are some oscillations, indicating occasional missed detections.

The mAP_{0.5} graph depicts the model's average precision at an IoU threshold of 0.5. After an initial growth spurt, the mAP_{0.5} stabilizes and fluctuates around a high value, suggesting robust object detection performance for this specific overlap criterion. For the mAP_{0.5:0.95} consistent growth in the initial epochs followed by a plateau is observed. Achieving a high value here implies that the model performs well not just at one IoU threshold but across many, demonstrating its versatility in detecting objects with varying levels of overlap accuracy.

In conclusion, the YOLOv5 model demonstrates significant training progression, as evident from the presented metrics. The precision, recall, and mAP scores, showcase a well-trained model capable of accurate object detection across various scenarios. To prevent overfitting, early stopping regularization technique was employed in this study during training. It involves monitoring the validation loss and halting the training process when this metric cease to

improve for a specified number of consecutive epochs, termed "patience." Concurrently, the model's weights are saved at each epoch where the validation metric exhibits improvement. Upon reaching the stopping criterion, the model's weights are reverted to the state where the validation performance was optimal, ensuring that the finalized model represents its best-performing state on unseen data.

4.2.6 Industrial Use & Limitations

Each spray torch possesses unique characteristics, and the model should be trained based on the specific equipment used. New image capture and annotations should be performed, including any other classes of interest. The model can be easily implemented by converting the YOLOv5 model to a format suitable for mobile devices. The mobile used is a flagship device using Android 11. The chipset is an Exynos 2100 (5 nm), Qualcomm SM8350 with a CPU Octa-core (1x2.9 GHz Cortex-X1 & 3x2.80 GHz Cortex-A78 & 4x2.2 GHz Cortex-A55). The build-in GPU is a Mali-G78 MP14. Various frameworks are available for converting the PyTorch YOLOv5 model to a format compatible with mobile devices, such as TensorFlow Lite, ONNX, or Core ML. Due to the limited GPU resources on mobile devices, the YOLOv5n model can offer fast performance with image processing speeds as low as 45ms (Table 4.3). Alternatively, high-end GPUs can be used on-site to perform tasks using the larger and more accurate YOLOv5 models. Mainstream cameras delivering high FPS at a resolution higher than 640px would be required in this case. Finally, object detection and classification can be done remotely through streamlined videos.

The operator can observe the torch operating status in real time, as shown in Figure 4.9. The areas of interest appear as anchor boxes, with confidence levels displayed next to the predicted class. In case of process deviation from optimal spray conditions, the model makes no predictions, or the confidence level for each prediction is significantly reduced. This allows the

operator to stop and inspect the torch and peripherals affecting gun operation. Historic data of detected classes for a specific spray job can be used as an additional QA/QC safety net. Of all tested models, the best balance between performance and accuracy has been achieved by the YOLOv5s model. The coating microhardness results are experimentally verified, as all unseen validation and test samples in the dataset were annotated based on measured coating microhardness levels of sprayed samples. The gas input parameters were cross-validated using a fully calibrated mass flow control unit.

Each spray parameter, as displayed in Table 4.1, results in a class group that emerges concurrently as classified anchor boxes. For example, when spray parameters A are chosen from the control panel, classes 1, 3, 4, and 7 appear together, forming Class Group 1. If a class is absent from the image, it indicates that the model was unsuccessful in locating and classifying the anchor box. If more than four classes are detected, it suggests that the model confidently predicts two classes belonging to distinct spray parameters. If four classes are present in an image, and one or more belong to a different group, it signifies that the model failed to accurately predict one or more classes, resulting in imprecise predictions. In practice, any variations in spray conditions will manifest in one or more of the aforementioned forms in the output file.

To address the need for a more qualitative analysis of the monitoring process and to identify any limitations, the number of missing classes from each group is plotted against each video frame during spraying, as shown in Figure 4.10. This is a classification analysis frame-by-frame showing the model behavior under different spray parameters, view angles and distances from the torch. Class group 1 includes classes 1, 3, 4 and 7. Class group 2 includes classes 2, 5, 6 and 8. Frames 0-200 correspond to view angle 1 and spray parameters A. Frames 200-260 correspond to compromised angle view 2 and spray parameters A. Frames 260-450 correspond to view angle 3 and spray parameters B as shown in Table 4.1. Figure 4.10 can be interpreted

as following: when 4 classes from group 2 are missing, then there is 100% confidence that the torch operates with parameters A and vice versa. When only 1 parameter is missing, then it can be safely deduced which set of parameters (A or B) the torch operates. In the case of 2 parameters missing from the same parameter group, then the torch status is undetermined. The scenario representing ideal operating conditions would result in a clear separation of class groups 1 and 2, as demonstrated for frames 0-200 and 260 - 450, respectively. In this case, the camera is positioned at a short distance with a full side view of the jet. The model accurately predicts the Group 1 classes, with no missing or misclassified anchor boxes. As expected, all Group 2 classes are absent from the predictions for all frames from 1 to 200. When the view angle is severely compromised (view angle 2) for the same spray parameters, the model generates confusing predictions, where Group 1 and Group 2 classes are mixed up from frames 200 to 260. Additionally, several frames display a reduced number of anchor boxes, indicating that the model failed to detect the regions of interest at this view angle. When the spray parameters were changed to parameters B, from frames 260 to 450, Group 2 classes were once again clearly separated from Group 1 classes.

On only one occasion (between frames 300-320), the model failed to detect a class in Group 2; however, since all other three classes were successfully detected and no Group 1 classes were detected, these instances had no impact on the operating status confidence. In practice, the model performed exceptionally well when the camera was positioned at any distance from the torch and for jet being perpendicular to the camera optical axis. The model's predictions deteriorate when the jet becomes severely oblique to the camera axis. YOLO is a single-stage detector that uses a grid-based approach to detect objects. This means that it divides the image into a grid of cells and predicts bounding boxes and class probabilities for each cell. When the jet becomes severely oblique to the camera axis, it may not be fully contained within a single cell, which could lead to inaccurate predictions. The automated frame analysis can be

employed to monitor the torch state for extended operating periods. When the pattern changes and the group classes begin to mix, this serves as a strong indication that maintenance is needed to prevent coating conformance issues. Any sudden fluctuation due to leaks or nozzle clogging results in pattern changes, as depicted in Figure 10.

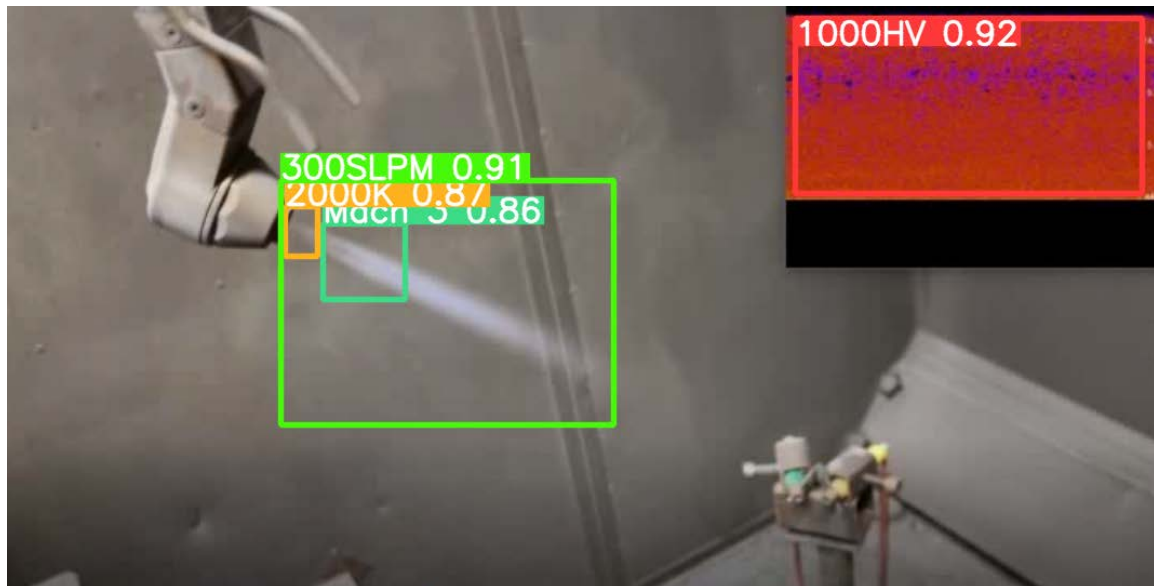


Figure 4.9 Detected anchor boxes and predicted process parameters with confidence levels.

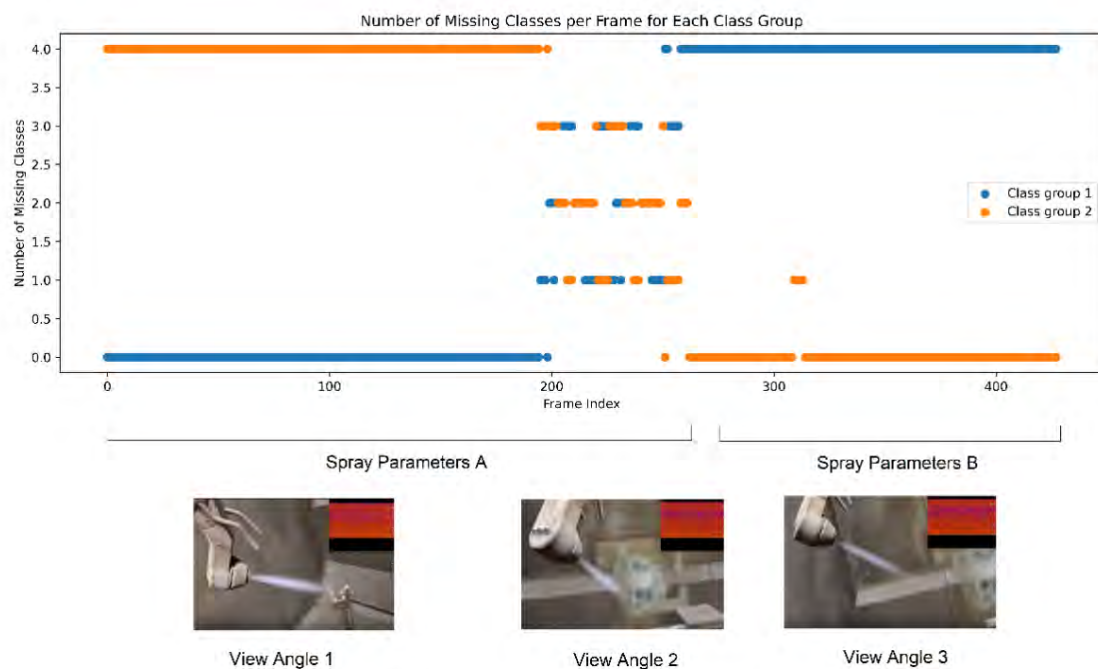


Figure 4.10. Classification analysis frame-by-frame showing the model behavior under different spray parameters, view angles and distances from the torch.

Model	Image Size	F1	Confidence	mAP_0.5:0.95	Detection Speed (GPU-RTX A5000)	Detection Speed (Mobile Phone)
YoloV5x	640	0.95	0.753	0.6	21.1ms	766ms
YoloV5l	640	0.9	0.59	0.58	17.4ms	430ms
YoloV5m	640	0.95	0.705	0.55	13.9ms	224ms
YoloV5s	640	0.95	0.702	0.59	11ms	98ms
YoloV5n	640	0.96	0.651	0.6	10ms	45ms

Table 4.3 Model accuracy and performance comparisons

4.2.7 Conclusions

This study demonstrates the successful implementation and evaluation of the YOLOv5 object detection architecture for real-time monitoring of thermal spray process parameters. The research encompassed a thorough examination of the YOLOv5 model family, including YOLOv5n, YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. Each model was benchmarked and assessed using various performance metrics such as PR-curve, R-curve, F1-curve, and confusion matrix analysis. Apart from the assessment of the YOLOv5 models, our work introduces a new approach to predicting coating quality by combining acoustic emissions and process images. This integration of two different data sources allows for a more comprehensive understanding of the process and may lead to improved predictions of the final coating quality.

The fusion of acoustic emissions and process images in this study takes advantage of the complementary information provided by each data source. Acoustic emissions offer insights into the dynamic behavior of the thermal spray process, while process images provide detailed visual information on the spray parameters. The results indicate that the YOLOv5 models perform exceptionally well in detecting and classifying process parameters in the thermal spray application. Notably, the YOLOv5s model provides the best balance between performance and accuracy, while the YOLOv5n model offers fast performance for mobile devices with limited GPU resources. Moreover, the coating microhardness results were experimentally verified and

the gas input parameters were cross-validated, providing further confidence in the performance of the models.

The real-time monitoring enabled by the YOLOv5 models facilitates improved process control, allowing operators to promptly identify deviations from optimal spray conditions, inspect the torch and peripherals, and make necessary adjustments. The use of historical data as an additional QA/QC safety net further enhances the reliability of the system.

In summary, the application of YOLOv5 models for thermal spray process parameter detection and classification presents a promising approach for improving process control and ensuring the quality of coatings produced. All models achieve high precision and recall while the obtained average accuracy was above 90% for all classes. The smallest model YOLOv5n can deliver detection speeds up to 100 frames per second on a high-end GPU and 20 frames per second using a modern smartphone. The model can be used to detect in real-time any deviation from the optimal spray parameters. This can be done in a dynamic manner while the torch is in motion. The model is capable of delivering accurate results that could be of enormous value to industrialists seeking a flexible and cost-effective method to monitor their equipment and process. Future research may explore the integration of these models with other monitoring techniques currently available in the market.

Chapter 5. Post Spray Quality Control and Microstructural analysis

5.1 Methodology

The methodology for "Post Spray Quality Control" in Section 5.2 focuses on advanced computer vision and machine learning techniques for microstructural characterization and analysis of spray-coated materials, particularly in thermal spray shops. This section outlines the application of these technologies in quality control processes, specifically addressing the challenges in traditional manual inspection methods. In this section the author highlights the necessity for digital methods in quality assurance (QA/QC) processes to enhance productivity, minimize human error, and reduce costs, thereby keeping small and large spray shops competitive. The key aspects include:

1. **Structural Image Similarity for Coating Conformance:** Utilizes structural image similarity (SSIM) to quantify coating conformance and quality. The SSIM index is used to compare microstructural features of coatings and establish quality metrics. A higher SSIM index indicates better conformance to a reference sample.
2. **Microstructure Feature Classification with Machine Learning:** Discusses the use of convolutional neural networks (CNNs) to classify whole images based on microstructural features such as oxides, porosity, and feedstock material types. The CNNs are trained on labeled datasets of coating cross-section images to classify different types of microstructural characteristics.
3. **Object Detection in Microstructure Analysis:** Use of EfficientNet for the automatic detection of coating structures such as oxides and porosity. A validation study demonstrates the model's ability to classify defects in micrographs accurately by locating and outlining individual microstructural features. This includes differentiating

between oxides and porosity and correlating these findings with chemical maps for validation.

4. **Training and Validation of the Models:** Describes the process of training and validating these models, including the analysis of training accuracy, validation accuracy, and loss functions. The models show promise in classifying microstructural features with high confidence levels, even with relatively small training datasets.
5. **Super Resolution Techniques in Microstructure Analysis:** Introduces the Efficient Sub-Pixel CNN (ESPCN) model for image super-resolution, enhancing the quality of microstructure images. This approach is shown to be superior to traditional interpolation methods like bicubic interpolation, providing clearer and more accurate images at high magnifications.
6. **Quantitative Analysis and Correlation Studies:** Suggests further experimental work to correlate the effects of image resolution on measured coating properties and assess the necessity of super-resolution techniques in coating inspection procedures.

In summary, this section emphasizes the integration of advanced computer vision and machine learning techniques in post-spray quality control for microstructural analysis. These methods aim to provide more objective, reliable, and efficient quality assessment tools for thermal spray coatings, potentially revolutionizing quality control processes in the industry.

5.2 Results & Discussions of Post Spray Quality Control

5.2.1 Image similarity index

In section 4 several computer vision and ML approaches were discussed for the microstructural characterisation and analysis of the spray coated materials. In thermal spray shops, that are accredited to process critical parts, the coating quality inspection process is integral part of production with pre-spray and post-spray checks as per OEM requirements. At the moment these operations are very labour intense and to some extent problematic when it comes to subjective human expert judgement. Practical solutions to accelerate the QA/QC processes that will allow small and large spray shops to remain competitive are necessary. The digital transformation of quality procedures will inevitably boost productivity, minimise human error and reduce costs allowing small and large spray shops to remain competitive in global markets. There is no doubt that the advancement of digital microscopy equipment will go hand in hand with computer vision techniques coupled with machine learning algorithms. How this will be done remains a challenge without clear guidance from coating experts and analysis of the current sector needs.

The recent advances in structural image similarity offer an excellent opportunity to address the challenge of quantitative measure of coating conformance and quality. In the context of spray coated samples, the method can be used to compare the microstructural features of coatings and establish conformance quality metrics. To validate this approach the author, presents in Appendix C the implementation code of the state-of-the-art structural similarity index function.

The coatings shown in Table 5.2 were HVOF thermally sprayed using the Cantor High Entropy Alloy (HEA) as feedstock material. The samples were produced using different spray conditions as shown in the same table. The samples were cut, polished and the images were

captured using the SEM method. The purpose of this comparison is to assess the ability of the SSIM algorithm to provide meaningful quality and coating conforming metrics.

As shown in Table 5.2 the SSIM index is higher when a conforming coating is compared to a reference sample. This example demonstrates that this method can be used to quantify the coating microstructure quality when compared to an optimised conforming coating. The key characteristic of the SSIM index is the ability to produce independent, objective and repeatable results. This enabling tool can offer new approaches to microstructural analysis, including the development of new visual metrics and the quantification of processing-microstructure-property relationships.

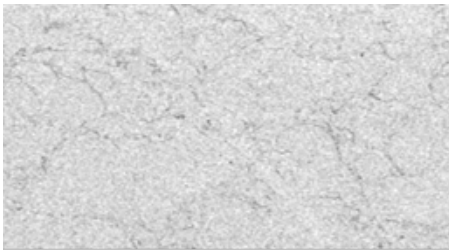
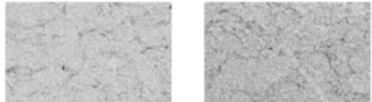
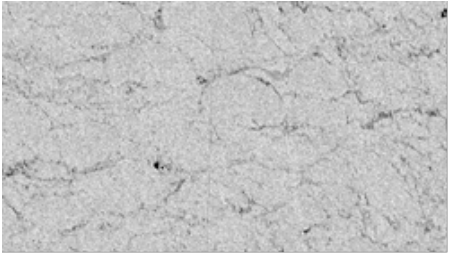
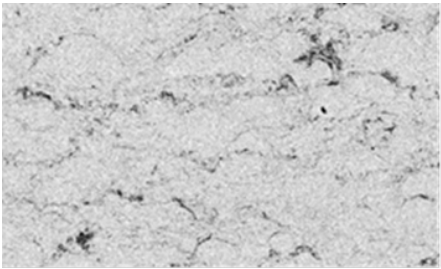
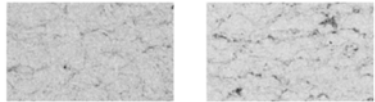
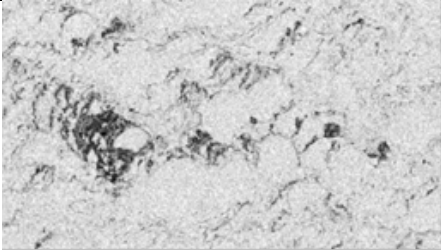
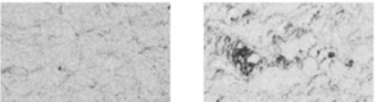
Sample	Description	Microstructure (SEM)	Structural Similarity (SSIM) Index
HEA 1	Optimised sample		 0.14
HEA 2	Reference Conforming Coating		Reference Image
HEA 3	Medium temperature spray showing medium oxide content		 0.11
HEA 4	High temperature spray showing high oxide content		 0.10

Table 5.2. Coating microstructure comparisons using the SSIM index as a quality metric. The optimised coating HEA1 shows the best structural similarity index with the reference conforming coating, while the compromised coatings with high oxide content have a lower SSIM index score. The worst coating has the lowest SSIM index of all samples.

5.2.2 Cross sectional image classification

The literature also suggests that microstructure feature classification can be performed using ML with very high accuracy [145]. The convolutional neural networks (CNNs) could be trained to classify images based on their microstructural features such as oxides, porosity,

unmelts, cracks, grit entrapment and type of feedstock material used to produce a coating. For the validation study, different crosssectional images were gathered and labelled according to their oxide and porosity levels. A CNN with the structure showing in Figure 5.10a and in Appendix D was trained on the images to identify the presence of porosity and oxides in a coating and classify any given images into three categories (Class 1: High Oxides, Class 2: High Porosity and Class 3: Conforming).

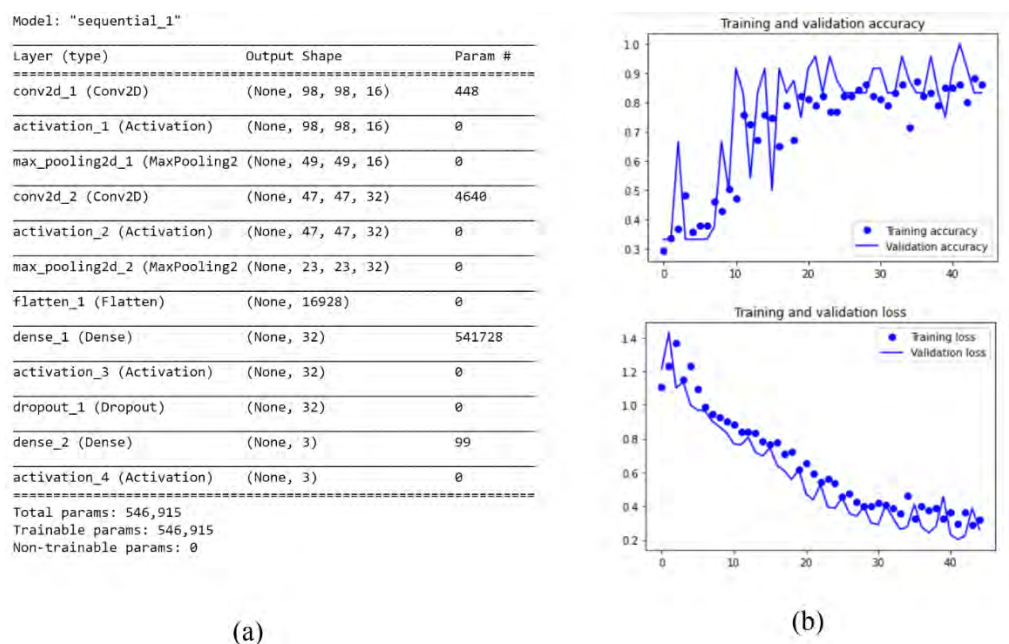


Figure 5.10. a) Model summary and architecture, b) Training and validation accuracy and loss for 40 epochs of training.

A database of 94 cross section images were collected to train the CNN model and 24 images for validation purposes. The images gathered were reduced in size before being used in the CNN. Several input sizes were tested and the minimum image resolution of 100×100 px found to have the highest accuracy utilising the minimum computational resources. Many different architectures were tested and models with different complexity levels were used to train for the coating images, and the results were compared to find the most effective yet light model that could identify the presence of porosity and oxides. It was discovered that the effectiveness of

a simple model with 3 layers of convolution with 3x3 receptive fields, were just as effective as larger and more complex models. All layers, except for the final layer have ReLu activation functions, while the last layer has a SoftMax activation function. There are three convolution layers followed by max pooling layers after each convolution layer, and after the third max-pooling layer a fully connected layer exists followed by another layer, with a drop out of density 0.12 placed between the two, and finally the model ends with a SoftMax layer with the size being set to three which is the number of categories at the output. For the training of the model itself, the Adam optimisation technique was used with sparse categorical cross entropy loss function.

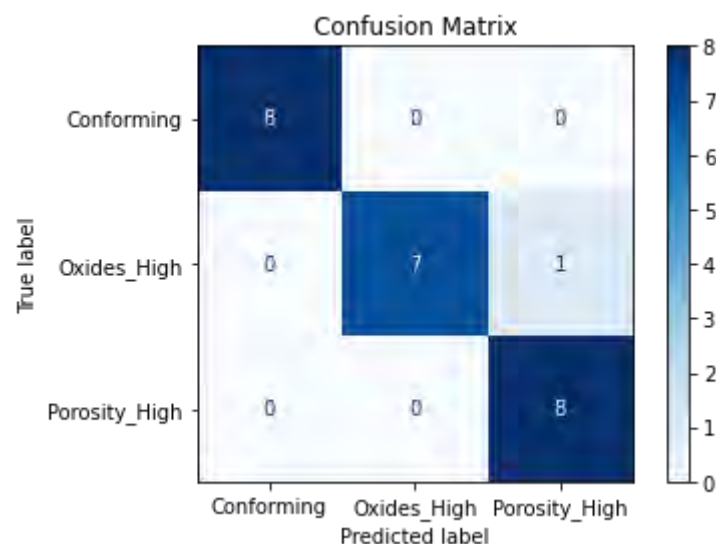


Table 5.3 The confusion matrix for the test subset for image classification

As shown in Figure 5.10.b and Table 5.3, the most accurate model after the training for this dataset had an 95% accuracy on the training set and was 95% accurate on the validation set, which implies that it can be expected that this model would be very accurate in classifying any image that falls under the three categories the model was trained for. Random unseen test input images, activation maps and classification probabilities are shown in Table 5.4. It is evident that the model is trained correctly focusing on microstructural features to classify the coating microstructural properties. Taking into account the relatively small training dataset, the model

can be further improved with higher confidence scores for all classes. The same approach can be used to classify any coating defect provided that sufficient data are available for the additional classes in the model.

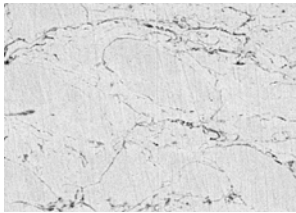
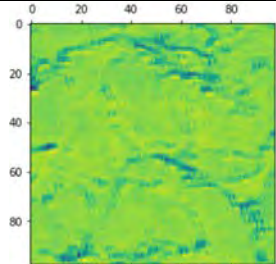
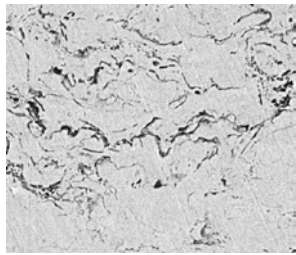
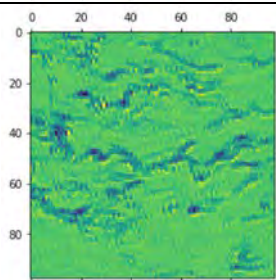
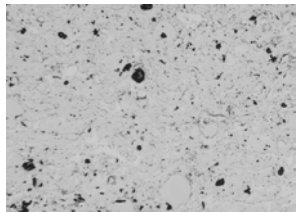
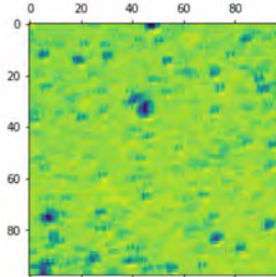
Description	Unseen Data (Microstructure Images)	Activation Map (4 th channel of the activation of the 1st layer of the original model)	Class Prediction & Confidence Levels
Cantor High Entropy Alloy Microstructure via Low Temperature HVOF: Showing Low Oxides and Porosity.			Conforming: 83% High Porosity: 16% High Oxides: 1%
Cantor HEA Coating Sprayed Under High Flame Temperature Conditions: Microstructure Showing High Oxide Content			High Oxides: 97% High Porosity: 3% Conforming: 0%
Coating Showing High Porosity Levels			High Porosity: 79% High Oxides: 15% Conforming: 6%

Table 5.4. Model validation using HVOF spray coated images that have not been used to train the model. Three representative coatings are classified based on their microstructural characteristics. All three classes were predicted with high levels of confidence.

5.2.3 Object detection in cross sectional images

The next challenge regarding the microstructural analysis is the automatic detection of coating structures. Object detection entails locating each unique microstructural feature in an image, i.e., finding each individual pore or oxide in a micrograph. In typical thermal spray cross

sections, the features are most of the times spatially separated and object detection can be performed using a conventional image analysis approach such as the connected-component labelling. In this case transfer learning allows these systems to be trained on natural images (such as the COCO detection data set) and applied successfully to microstructural images. For example, in order to assess the prevalence of porosity and oxides in a thermally sprayed coating, the first task is to identify the discrete pores and oxide clusters in a set of SEM or optical microscopy images. Then tedious manual annotation needs to be done for each image and use them to train a CNN instance detection and classification system.

This approach is highly recommended for further research and a validation study is presented in this section based on a state-of-the-art model EfficientNet. The main building block for EfficientNet is the MBConv using shortcuts between bottlenecks and is combined with in-depth separable convolutions that can reduce the calculation by almost k^2 where k is the height and width of the 2-D convolution window (Kernel Size). For demonstration purposes a relatively small dataset consisting of 40 micrographs (32 for training and 8 for testing) belonging to two main classes (Class 1: Oxides and Class 2: Porosity) was collected and the individual features were labelled manually. Example coating microstructures for the two classes used in the classification test are shown in Figure 5.11.

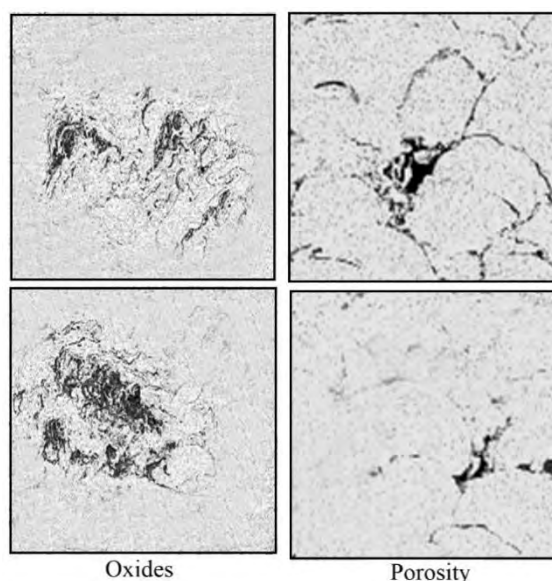


Figure 5.11. Example coatings with Oxide and Porosity structures used for labelling and classification training and testing. The images have been processed in an effort to eliminate training confusion from the surrounding structures.

The classifier achieves a validation accuracy of approximately 70% after 3000 epochs as shown in Tensorboard training histograms Figure 5.12. The loss is calculated on training and test and its interpretation is based on how well the model is doing in these two sets. It is the sum of errors made for each example in training or validation sets. Loss value implies how poorly or well a model behaves after each iteration of optimization. Due to the small dataset size, no efforts have been made to fine-tune the model hyper-parameters such as the SVM regularization parameter, which would require an additional validation circle. Nonetheless, these results demonstrate the potential for deep learning machine vision approaches to microstructural classification. Even using untuned model parameters and a small training set, the system achieves accurate classification performance without human intuition. This is of course lower than the performance of state-of-the-art recognition systems with performance in the range of 90%. With further optimization and larger datasets, we are confident that this method will earn its academic and commercial merit in the future high performance quantitative and qualitative microstructure applications.

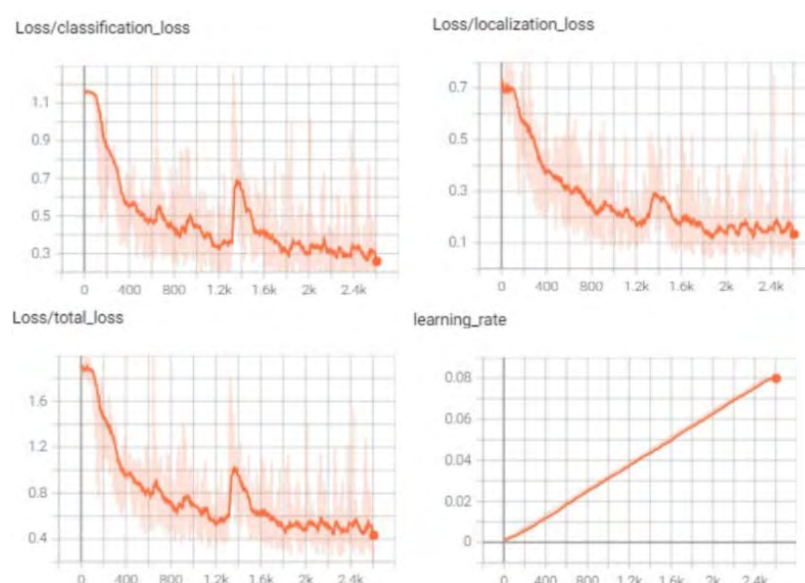


Figure 5.12. Detection and classification model training histograms (TensorBoard)

In some cases, it is arguable whether the computer or human is correct. For this reason, the oxygen chemical map has been used alongside the object detection predictions to assess whether defected areas correspond to oxides or porosity as shown in Figure 5.13. The model was successful at locating and outlining bounding boxes around individual microstructural features, achieving a high detection recall. Loss of recall occurs for very small and widely spread defects. In this case a large box is delineated by the model to indicate global defects as opposed to local flaws. More specifically as shown in Figure 5.13.a, the model correctly identifies the oxide cluster and classifies this area as oxide rich. Similar results were found from the EDS analysis shown below. In Figure 5.13.b the oxide cluster is correctly identified but the system didn't find any porosity within the oxide cluster as expected. The individual pores in Figure 5.13.c are clearly detected in the splat boundaries. Finally, the example in Figure 5.13.d shows a more complicated structure where oxides and porosity co-exist in the microstructure. The model draws two overlapping boxes over the two defected sites suggesting that oxides and porosity are both present. A human expert would conclude the same in this case. Overall, although the confidence levels are low the model correctly points into the

defected areas with reasonable classification accuracy. Clean areas are not miss-classified or confused with defects. The errors that do occur for example in Figure 5.13.b with the prediction of porosity over the oxide cluster are sensible and to some extent unavoidable without model optimisation and large dataset training. In summary, this class of ML models can capture the meaningful details and defining characteristics of coating microstructural images without explicit fine-tuning from human experts. This constitutes a paradigm for microstructure analytics, which complements the current research emphasis on quantitative characterization and design of particular microstructural systems. This approach to microstructure data is flexible enough to span across different spray coated materials and methods and appears robust enough to enable automatic search and accurate classification of large groups of microstructural image data.

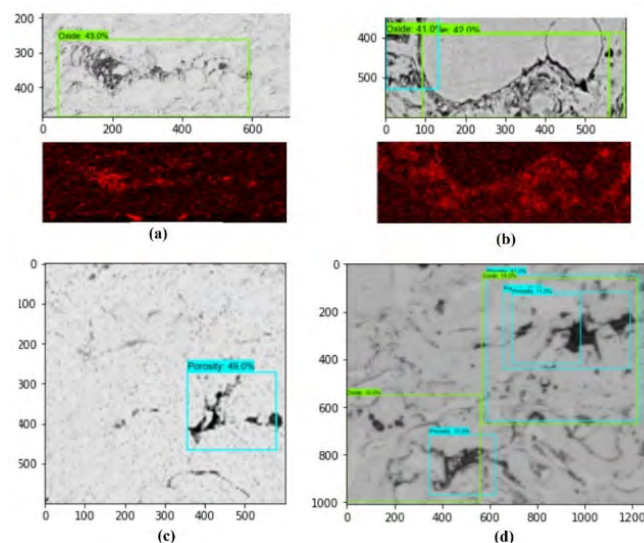


Figure 5.13. Object identification and instance segmentation for coating micrographs via thermal spray. The green boxes correspond to oxides and the light blue to porosity as predicted by the EfficientNet. (a) The image contains oxide clusters and the corresponding oxygen map is shown below. (b) Oxide clusters and possible porosity around an unmelt particle with oxygen map below. (c) Porosity formation from poor lamella formation. (d) Microstructure showing both porosity and oxides.

5.3 Results & Discussions of Microstructural Analysis Accuracy and Improvements

In this section, the author validates and proposes the use of deep learning architectures for upscaling an already acquired image, in order to assess quantification and detection of spray coated microstructures. The rationale behind this, is that ideally images of very high resolution, (such as those produced by Scanning Electron Microscopy -SEM) are required for this task, whereas most enterprises use optical microscope images. More specifically, this research focused on utilizing ESPCN (Efficient Sub-Pixel CNN) models [174] for increasing the pixelation of spray coated microstructures. We assume that the model should be trained to learn the array of upscaling filters instead of using a super resolution algorithm trained in non-digital microstructure related datasets. In this model the handcrafted bicubic filter in the super resolution pipeline is replaced by a more complex upscaling filter and is specifically trained for a coating microstructure feature map in this thesis. Interpolation methods typically used to accomplish the task, such as bicubic interpolation, do not bring additional information to solve the ill-posed reconstruction problem.

The training of upscale filters has been conducted using 90 random spray coated microstructural images. The dataset was split into training (80%), validation (10%) and testing (10%) subsets. In the training phase, sub-images are extracted from the training ground truth images multiplied by r , where r is the upscaling factor that was set to 8 in this case. To synthesize the low-resolution samples, the high-resolution images are blurred using a Gaussian filter and sub-sampled by the upscaling factor (Appendix E).

Examples of the reconstructed microstructure images from low-resolution inputs are shown in Figure 5.14.a and the resolution enhancement during training after 1,5 and 25 epochs in Figure 5.14.b, indicating that the regeneration was completed successfully. In Figure 5.15, high resolution microstructures are further enhanced by a factor of 8 and compared against bicubic

interpolation image upscaling. Clearly the ML approach produces sharper images without significant loss of information at high magnification as opposed to conventional methods.

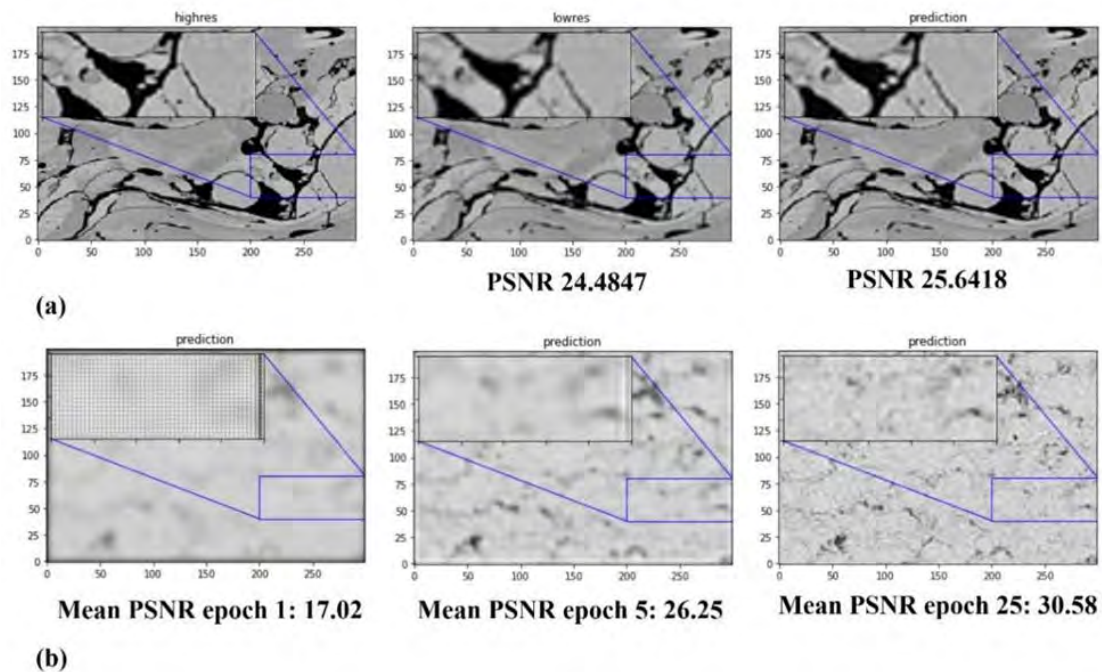


Figure 5.14. (a) Image reconstruction from a low-resolution image and comparison with the high-resolution ground truth image on the left. The PSNR has increased after image upscale. (b) Images showing resolution enhancement during model training and the equivalent increase of PSNR.

Finally, ImageJ™ is used to measure and compare the porosity between a super resolved and the original image. It is evidently shown that there are differences in the calculated porosity up to 0.5%. One can see that with the decreasing resolution, porosity boundaries become increasingly pixelated at high magnification. As a result, the total calculated dark area in the binary image is larger compared to the smoother super-resolved image. In this context experimental work should be carried out to correlate the image resolution effect on the measured coating properties and to assess the necessity of super resolution formalisation in coating inspection procedures.

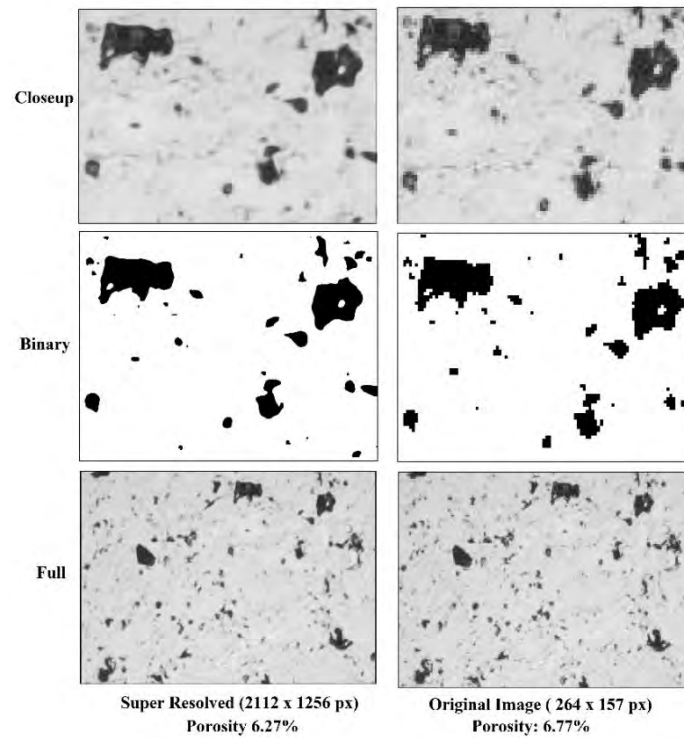


Figure 5.15. Spray coated microstructure images super resolved using deep convolution neural networks. The high-resolution images on the left were generated by upsampling the lower resolution image on the right with the use of CNN super resolution techniques.

Chapter 6 Conclusions

The spray process journey in a coating service centre starts with the analysis of the component to be sprayed and its distinct geometrical features. The first key challenge practitioners are faced with is the coating conformity as per customer specifications and the suitability of kinetic spray deposition as an application option overall. Modelling and simulations tools can be used to map the coating properties on the surface of the part eliminating in such a way the need for destructive testing and long iterative and experimental spray process optimisation. Available technologies are based on semi-analytical models with good scalability, where the entire spray coating process of a component can be simulated in short time. This includes the visualisation of the coating evolution in a virtual environment and digital post examination of coating properties as an alternative to destructive testing. Nevertheless, these models are limited to the assumption of a relatively small addition of material on the surface of the component. This approach is not suitable for additive manufacturing where the coated material alters the topology of the surface significantly and more research is needed in this direction.

In relation to application optimisation through automated robot planning, several approaches have been discussed and their advantages and limitations were highlighted. The ML driven Self-Organising-Map method appears very promising delivering excellent results when compared to a human expert. The current main limitation of this approach is the compromised performance when used with non-axisymmetric 3D objects. More research is required for the development of a complete multidimensional model.

The next critical step, moving into production, is the process monitoring and control by means of closed loop systems and sensing technologies. The current state-of-the-art is discussed in the third section of this work where the necessary developments required to address current challenges and the limitations of existing systems are highlighted. Due to the static nature of

commercial optical instruments, continuous process monitoring is trivial and, in many cases, impractical or impossible. A typical example is the application of thermal spray coatings on the internal surface of parts where there is no visibility in confined areas. These applications require a new approach for process monitoring and control. The most promising technology is based on the analysis of airborne acoustic emissions in the booth using convolution neural networks. The same approach can be used for the classification of spray process images where minimal calibration is required in the presence of large datasets. The implementation of the algorithm is straightforward as shown in section five but more research efforts are needed for the generation of large datasets, kernels and feature vectors.

In this direction this work demonstrates the successful implementation and evaluation of the YOLOv5 object detection architecture for real-time monitoring of thermal spray process parameters. The research encompassed a thorough examination of the YOLOv5 model family, including YOLOv5n, YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. Each model was benchmarked and assessed using various performance metrics such as PR-curve, R-curve, F1-curve, and confusion matrix analysis. Apart from the assessment of the YOLOv5 models, our work introduces a new approach to predicting coating quality by combining acoustic emissions and process images. This integration of two different data sources allows for a more comprehensive understanding of the process and may lead to improved predictions of the final coating quality.

The fusion of acoustic emissions and process images in our study takes advantage of the complementary information provided by each data source. Acoustic emissions offer insights into the dynamic behavior of the thermal spray process, while process images provide detailed visual information on the spray parameters. The results indicate that the YOLOv5 models perform exceptionally well in detecting and classifying process parameters in the thermal spray application. Notably, the YOLOv5s model provides the best balance between

performance and accuracy, while the YOLOv5n model offers fast performance for mobile devices with limited GPU resources. Moreover, the coating microhardness results were experimentally verified and the gas input parameters were cross-validated, providing further confidence in the performance of the models.

The real-time monitoring enabled by the YOLOv5 models facilitates improved process control, allowing operators to promptly identify deviations from optimal spray conditions, inspect the torch and peripherals, and make necessary adjustments. The use of historical data as an additional QA/QC safety net further enhances the reliability of the system.

In summary, the application of YOLOv5 models for thermal spray process parameter detection and classification presents a promising approach for improving process control and ensuring the quality of coatings produced. All models achieve high precision and recall while the obtained average accuracy was above 90% for all classes. The smallest model YOLOv5n can deliver detection speeds up to 100 frames per second on a high-end GPU and 20 frames per second using a modern smartphone. The model can be used to detect in real-time any deviation from the optimal spray parameters. This can be done in a dynamic manner while the torch is in motion. The model is capable of delivering accurate results that could be of enormous value to industrialists seeking a flexible and cost-effective method to monitor their equipment and process. Future research may explore the integration of these models with other monitoring techniques currently available in the market.

In the final production step, when thermal spray deposition is complete, the parts are thoroughly checked for quality conformance. In this last production step, a human expert assesses the product and raises conformance or non-conformance reports as appropriate. The fourth section of this review article focuses on current quality procedures, best practises, limitations and future prospects. The author present and discuss methods for image quality enhancement using CNNs from extensive literature reviews. It is demonstrated that the image

quality has an effect on coating property measurements and that more research is needed in this direction to compare super resolved image characterisation with advanced experimental methods beyond SEM and optical microscopy. From the modelling and simulation point of view, the literature suggests that super resolved digital images can be of enormous value. The key requirement for further utilisation of this approach is the collection of large coating microstructure datasets that would be used to train the super resolution algorithms. The design of new feature vectors and Kernels based on material science theory can lead to predictive microstructural analysis through super resolution for scales currently inaccessible by state-of-the-art equipment.

In the same direction, the literature review suggests that a structural similarity index can be obtained and used for coating microstructure comparisons. Automating this very labour intense process, that is primarily based on subjective human expert judgement, can open up new standardisation avenues. Will inevitably boost productivity, minimise human error and reduce costs. Training more models with coating microstructural features will be the key to developing this technology further including the ability to fully automate the coating defect detection using CNNs.

In section five, the author presents the implementation steps of the identified key enabling ML approaches. The main objective is to address in practise some key challenges and to demonstrate that the obstacles standing in the way of widespread adoption of machine learning in the field of thermal spray technologies are surmountable. This includes tackling the uncertainty about what machine learning can do, the challenge of data preparation, the lack of public labelled datasets and finally the lack of kernels and features specifically designed to enhance the performance of existing models for surface engineering applications.

List of References

- [1] T. Van Hoof, G. Fradet, F. Pichot, S. Selezneff, N. Poletz, Simulation of Thermal Spray Coating on 3D Objects: Numerical Scheme and Aeronautic Test Case, *J. Therm. Spray Technol.* (2019). <https://doi.org/10.1007/s11666-019-00933-6>.
- [2] C. Chen, H. Liao, G. Montavon, S. Deng, Nozzle Mounting Method Optimization Based on Robot Kinematic Analysis, *J. Therm. Spray Technol.* (2016). <https://doi.org/10.1007/s11666-016-0429-1>.
- [3] B. Zhang, J. Wu, L. Wang, Z. Yu, P. Fu, A Method to Realize Accurate Dynamic Feedforward Control of a Spray-Painting Robot for Airplane Wings, *IEEE/ASME Trans. Mechatronics.* (2018). <https://doi.org/10.1109/TMECH.2018.2817884>.
- [4] ABB. RobotStudio. Available online: <https://new.abb.com/products/robotics/robotstudio> (accessed on 31 October 2021), (n.d.).
- [5] D. Hegels, T. Wiederkehr, H. Müller, Simulation based iterative post-optimization of paths of robot guided thermal spraying, *Robot. Comput. Integr. Manuf.* (2015). <https://doi.org/10.1016/j.rcim.2015.02.002>.
- [6] Z.J. Shi, J. Guo, A new algorithm of nonlinear conjugate gradient method with strong convergence, *Comput. Appl. Math.* (2008). <https://doi.org/10.1590/s1807-03022008000100006>.
- [7] R. Gadow, A. Candel, M. Floristán, Optimized robot trajectory generation for thermal spraying operations and high quality coatings on free-form surfaces, *Surf. Coatings Technol.* (2010). <https://doi.org/10.1016/j.surfcoat.2010.08.121>.
- [8] D. Fang, S. Deng, H. Liao, C. Coddet, The effect of robot kinematics on the coating thickness uniformity, in: *J. Therm. Spray Technol.*, 2010. <https://doi.org/10.1007/s11666-010-9470-7>.
- [9] S.H. Deng, Z.H. Cai, D.D. Fang, H.L. Liao, G. Montavon, Application of robot offline programming in thermal spraying, *Surf. Coatings Technol.* (2012). <https://doi.org/10.1016/j.surfcoat.2012.03.038>.
- [10] Z. Cai, S. Deng, H. Liao, C. Zeng, G. Montavon, The effect of spray distance and scanning step on the coating thickness uniformity in cold spray process, *J. Therm. Spray Technol.* (2014). <https://doi.org/10.1007/s11666-013-0002-0>.
- [11] Z. Yanjun, L. Wenbo, L. Dayu, X. Jinkun, Z. Chao, Modeling of Thickness and Profile Uniformity of Thermally Sprayed Coatings Deposited on Cylinders, *J. Therm. Spray Technol.* (2018). <https://doi.org/10.1007/s11666-017-0661-3>.
- [12] Z. Cai, H. Liang, S. Quan, S. Deng, C. Zeng, F. Zhang, Computer-Aided Robot Trajectory Auto-generation Strategy in Thermal Spraying, *J. Therm. Spray Technol.* (2015). <https://doi.org/10.1007/s11666-015-0282-7>.
- [13] S. yu Ding, P. fei He, G. zheng Ma, S. ying Chen, H. dou Wang, Z. yong Wu, L. Tang, X. Chen, B. shi Xu, Numerical Simulation and Experimental Study of Heat Accumulation in Cylinder Parts During Internal Rotating Plasma Spraying, *J. Therm. Spray Technol.* (2019). <https://doi.org/10.1007/s11666-019-00908-7>.

- [14] S. Kamnis, S. Gu, Numerical modelling of droplet impingement, *J. Phys. D. Appl. Phys.* 38 (2005). <https://doi.org/10.1088/0022-3727/38/19/015>.
- [15] A. Kout, T. Wiederkehr, H. Müller, Efficient stochastic simulation of thermal spray processes, *Surf. Coatings Technol.* (2009). <https://doi.org/10.1016/j.surfcoat.2008.11.035>.
- [16] F.I. Trifa, G. Montavon, C. Coddet, Model-based expert system for design and simulation of APS coatings, *J. Therm. Spray Technol.* (2007). <https://doi.org/10.1007/s11666-006-9000-9>.
- [17] S. Barradas, V. Guipont, R. Molins, M. Jeandin, M. Arrigoni, M. Boustie, C. Bolis, L. Berthe, M. Ducos, Laser shock flier impact simulation of particle-substrate interactions in cold spray, *J. Therm. Spray Technol.* (2007). <https://doi.org/10.1007/s11666-007-9069-9>.
- [18] H. Zhou, C. Li, C. Bennett, H. Tanvir, C. Li, Numerical Analysis of Deformation Behavior and Interface Bonding of Ti6Al4V Particle After Subsequent Impact During Cold Spraying, *J. Therm. Spray Technol.* (2021). <https://doi.org/10.1007/s11666-021-01188-w>.
- [19] Y. Dou, J. Luo, L. Qi, H. Lian, X. Hou, Generation mechanism and suppression method of landing error of two successively deposited metal droplets caused by coalescence and solidification, *Int. J. Heat Mass Transf.* (2021). <https://doi.org/10.1016/j.ijheatmasstransfer.2021.121100>.
- [20] I. Malgarinos, N. Nikolopoulos, M. Marengo, C. Antonini, M. Gavaises, VOF simulations of the contact angle dynamics during the drop spreading: Standard models and a new wetting force model, *Adv. Colloid Interface Sci.* (2014). <https://doi.org/10.1016/j.cis.2014.07.004>.
- [21] Y.Z. Zheng, Q. Li, Z.H. Zheng, J.F. Zhu, P.L. Cao, Modeling the impact, flattening and solidification of a molten droplet on a solid substrate during plasma spraying, *Appl. Surf. Sci.* (2014). <https://doi.org/10.1016/j.apsusc.2014.08.032>.
- [22] M.S. Zoei, T. Farizeh, M.H. Sadeghi, M. Salehi, Residual Stress Distribution Through the Thickness of WC-10Co-4Cr Coating Deposited by HVOF: Experimental and Simulation Study, *J. Therm. Spray Technol.* (2020). <https://doi.org/10.1007/s11666-020-01012-x>.
- [23] V. Katranidis, S. Gu, B. Allcock, S. Kamnis, Experimental study of high velocity oxy-fuel sprayed WC-17Co coatings applied on complex geometries. Part A: Influence of kinematic spray parameters on thickness, porosity, residual stresses and microhardness, *Surf. Coatings Technol.* 311 (2017). <https://doi.org/10.1016/j.surfcoat.2017.01.015>.
- [24] A. Fardan, C.C. Berndt, R. Ahmed, Numerical modelling of particle impact and residual stresses in cold sprayed coatings: A review, *Surf. Coatings Technol.* (2021). <https://doi.org/10.1016/j.surfcoat.2021.126835>.
- [25] Z. Cai, X. Liang, B. Chen, C. Zeng, C. Chen, Z. Mu, T. Chen, A Geodesic-Based Robot Trajectory Planning Approach for Cold Spray Applications, *J. Therm. Spray Technol.* (2019). <https://doi.org/10.1007/s11666-019-00871-3>.
- [26] H. Wu, X. Xie, M. Liu, C. Chen, H. Liao, Y. Zhang, S. Deng, A new approach to

- simulate coating thickness in cold spray, *Surf. Coatings Technol.* (2020). <https://doi.org/10.1016/j.surfcoat.2019.125151>.
- [27] X.T. Luo, C.X. Li, F.L. Shang, G.J. Yang, Y.Y. Wang, C.J. Li, High velocity impact induced microstructure evolution during deposition of cold spray coatings: A review, *Surf. Coatings Technol.* (2014). <https://doi.org/10.1016/j.surfcoat.2014.06.006>.
 - [28] M.E. Lynch, W. Gu, T. El-Wardany, A. Hsu, D. Viens, A. Nardi, M. Klecka, Design and topology/shape structural optimisation for additively manufactured cold sprayed components, *Virtual Phys. Prototyp.* (2013). <https://doi.org/10.1080/17452759.2013.837629>.
 - [29] Q. Wang, N. Birbilis, M.X. Zhang, Process optimisation of cold spray Al coating on AZ91 alloy, *Surf. Eng.* (2014). <https://doi.org/10.1179/1743294413Y.0000000224>.
 - [30] V. Katranidis, S. Kamnis, S. Gu, Prediction of Coating Properties of Thermally Sprayed WC–Co on Complex Geometries, *J. Therm. Spray Technol.* 27 (2018). <https://doi.org/10.1007/s11666-018-0739-6>.
 - [31] M. Tzinava, K. Delibasis, B. Allcock, S. Kamnis, A general-purpose spray coating deposition software simulator, *Surf. Coatings Technol.* 399 (2020). <https://doi.org/10.1016/j.surfcoat.2020.126148>.
 - [32] T. MathWorks, MATLAB (R2020b), MathWorks Inc. (2020).
 - [33] T. Kohonen, Self-organized formation of topologically correct feature maps, *Biol. Cybern.* (1982). <https://doi.org/10.1007/BF00337288>.
 - [34] N. Kubota, Y. Nojima, F. Kojima, T. Fukuda, S. Shibata, Intelligent control of self-organizing manufacturing system with local learning mechanism, in: *IECON Proc. (Industrial Electron. Conf., 1999)*. <https://doi.org/10.1109/iecon.1999.816540>.
 - [35] A.A. Razavian, J. Sun, Cognitive based adaptive path planning algorithm for autonomous robotic vehicles, in: *Conf. Proc. - IEEE SOUTHEASTCON, 2005*. <https://doi.org/10.1109/secon.2005.1423236>.
 - [36] D. Miljkovic, Brief review of self-organizing maps, in: *2017 40th Int. Conv. Inf. Commun. Technol. Electron. Microelectron. MIPRO 2017 - Proc., 2017*. <https://doi.org/10.23919/MIPRO.2017.7973581>.
 - [37] V. Goncalves, K. Maria, A.B.F. da Silv, Applications of Artificial Neural Networks in Chemical Problems, in: *Artif. Neural Networks - Archit. Appl., 2013*. <https://doi.org/10.5772/51275>.
 - [38] E. Pampalk, S. Dixon, G. Widmer, Exploring music collections by browsing different views, *Comput. Music J.* (2004). <https://doi.org/10.1162/014892604323112248>.
 - [39] V.J.A.S. Lobo, Application of self-organizing maps to the maritime environment, in: *Lect. Notes Geoinf. Cartogr., 2009*. https://doi.org/10.1007/978-3-642-00304-2_2.
 - [40] W. E, Machine Learning: Mathematical Theory and Scientific Applications, *Not. Am. Math. Soc.* (2019). <https://doi.org/10.1090/noti1994>.
 - [41] M. Tzinava, K. Delibasis, S. Kamnis, Self-organizing Maps for Optimized Robotic Trajectory Planning Applied to Surface Coating, in: *IFIP Adv. Inf. Commun. Technol., 2021*. https://doi.org/10.1007/978-3-030-79150-6_16.

- [42] D. Ding, Z. Pan, D. Cuiuri, H. Li, A multi-bead overlapping model for robotic wire and arc additive manufacturing (WAAM), *Robot. Comput. Integr. Manuf.* (2015). <https://doi.org/10.1016/j.rcim.2014.08.008>.
- [43] O. Nenadl, W. Kuipers, N. Koelewijn, V. Ocelík, J.T.M. De Hosson, A versatile model for the prediction of complex geometry in 3D direct laser deposition, *Surf. Coatings Technol.* (2016). <https://doi.org/10.1016/j.surfcoat.2016.08.090>.
- [44] S. Suryakumar, K.P. Karunakaran, A. Bernard, U. Chandrasekhar, N. Raghavender, D. Sharma, Weld bead modeling and process optimization in Hybrid Layered Manufacturing, *CAD Comput. Aided Des.* (2011). <https://doi.org/10.1016/j.cad.2011.01.006>.
- [45] C. Chen, Y. Xie, C. Verdy, H. Liao, S. Deng, Modelling of coating thickness distribution and its application in offline programming software, *Surf. Coatings Technol.* (2017). <https://doi.org/10.1016/j.surfcoat.2016.10.044>.
- [46] H. Wu, X. Xie, M. Liu, S. Deng, H. Liao, A new approach to simulate coating thickness in cold spray, in: *Proc. Int. Therm. Spray Conf.*, 2019.
- [47] D. Kotoban, S. Grigoriev, A. Okunkova, A. Sova, Influence of a shape of single track on deposition efficiency of 316L stainless steel powder in cold spray, *Surf. Coatings Technol.* (2017). <https://doi.org/10.1016/j.surfcoat.2016.10.052>.
- [48] S. V. Klinkov, V.F. Kosarev, V.S. Shikalov, Influence of nozzle velocity and powder feed rate on the coating mass and deposition efficiency in cold spraying, *Surf. Coatings Technol.* (2019). <https://doi.org/10.1016/j.surfcoat.2019.04.004>.
- [49] W. Zhu, X. Zhang, M. Zhang, X. Tian, D. Li, Integral numerical modeling of the deposition profile of a cold spraying process as an additive manufacturing technology, *Prog. Addit. Manuf.* (2019). <https://doi.org/10.1007/s40964-018-0071-1>.
- [50] J. Xiong, G. Zhang, J. Hu, L. Wu, Bead geometry prediction for robotic GMAW-based rapid manufacturing through a neural network and a second-order regression analysis, *J. Intell. Manuf.* (2014). <https://doi.org/10.1007/s10845-012-0682-1>.
- [51] D. Ikeuchi, A. Vargas-Uscategui, X. Wu, P.C. King, Neural network modelling of track profile in cold spray additive manufacturing, *Materials (Basel)*. (2019). <https://doi.org/10.3390/ma12172827>.
- [52] M. Liu, Y. Zhang, W. Dong, Z. Yu, S. Liu, S. Gomes, H. Liao, S. Deng, Grey modeling for thermal spray processing parameter analysis, *Grey Syst. Theory Appl.* (2020). <https://doi.org/10.1108/gs-12-2019-0063>.
- [53] A. Olleak, Z. Xi, Calibration and Validation Framework for Selective Laser Melting Process Based on Multi-Fidelity Models and Limited Experiment Data, *J. Mech. Des. Trans. ASME*. (2020). <https://doi.org/10.1115/1.4045744>.
- [54] D. Ikeuchi, A. Vargas-Uscategui, X. Wu, P.C. King, Data-efficient neural network for track profile modelling in cold spray additive manufacturing, *Appl. Sci.* (2021). <https://doi.org/10.3390/app11041654>.
- [55] B.D. Sartwell, P.E. Bretz, HVOF thermal spray coatings replace hard chrome, *Adv. Mater. Process.* (1999).
- [56] B.E. Bodger, R.T.R. McGrann, D.A. Somerville, The Evaluation of Tungsten Carbide

Thermal Spray Coatings as Replacements for Electrodeposited Chrome Plating on Aircraft Landing Gear, *Plat. Surf. Finish.* (1997).

- [57] C. Moreau, Towards a better control of thermal spray processes, in: *Proc. Int. Therm. Spray Conf.*, 1998. <https://doi.org/10.31399/asm.cp.itsc1998p1681>.
- [58] C. Moreau, M. Lamontagne, P. Cielo, Method and apparatus for monitoring the temperature and velocity of plasma sprayed particles, CA2055267A1, 1991.
- [59] P. Fauchais, M. Vardelle, Sensors in spray processes, in: *J. Therm. Spray Technol.*, 2010. <https://doi.org/10.1007/s11666-010-9485-0>.
- [60] P. Fauchais, J.F. Coudert, M. Vardelle, A. Vardelle, A. Denoirjean, Diagnostics of thermal spraying plasma jets, *J. Therm. Spray Technol.* (1992). <https://doi.org/10.1007/BF02659011>.
- [61] J.R. Fincke, D.C. Haggard, W.D. Swank, Particle temperature measurement in the thermal spray process, *J. Therm. Spray Technol.* (2001). <https://doi.org/10.1361/105996301770349358>.
- [62] C.J. Li, T. Wu, C.X. Li, B. Sun, Effect of spray particle trajectory on the measurement signal of particle parameters based on thermal radiation, *J. Therm. Spray Technol.* (2003). <https://doi.org/10.1361/105996303770348537>.
- [63] J. Vattulainen, E. Hämäläinen, R. Hernberg, P. Vuoristo, T. Mäntylä, Novel method for in-flight particle temperature and velocity measurements in plasma spraying using a single CCD camera, *J. Therm. Spray Technol.* (2001). <https://doi.org/10.1361/105996301770349556>.
- [64] J.R. Fincke, W.D. Swank, R.L. Bewley, D.C. Haggard, M. Gevelber, D. Wroblewski, Diagnostics and control in the thermal spray process, *Surf. Coatings Technol.* (2001). [https://doi.org/10.1016/S0257-8972\(01\)01432-3](https://doi.org/10.1016/S0257-8972(01)01432-3).
- [65] M. Li, D. Shi, P.D. Christofides, Modeling and control of HVOF thermal spray processing of WC-Co coatings, in: *Powder Technol.*, 2005. <https://doi.org/10.1016/j.powtec.2005.04.011>.
- [66] M. Li, D. Shi, P.D. Christofides, Diamond Jet hybrid HVOF thermal spray: Gas-phase and particle behavior modeling and feedback control design, *Ind. Eng. Chem. Res.* (2004). <https://doi.org/10.1021/ie030559i>.
- [67] J. Ren, G. Zhang, Y. Rong, Y. Ma, A feature-based model for optimizing HVOF process by combining numerical simulation with experimental verification, *J. Manuf. Process.* (2021). <https://doi.org/10.1016/j.jmapro.2021.01.017>.
- [68] V. Srinivasan, A. Vaidya, T. Streibl, M. Friis, S. Sampath, On the reproducibility of air plasma spray process and control of particle state, in: *Proc. Int. Therm. Spray Conf.*, 2006. <https://doi.org/10.1361/105996306X147199>.
- [69] M.P. Planche, R. Bolot, C. Coddet, In-flight characteristics of plasma sprayed alumina particles: Measurements, modeling, and comparison, *J. Therm. Spray Technol.* (2003). <https://doi.org/10.1361/105996303770348555>.
- [70] S. Sampath, V. Srinivasan, A. Valarezo, A. Vaidya, T. Streibl, Sensing, control, and in situ measurement of coating properties: An integrated approach toward establishing process-property correlations, in: *J. Therm. Spray Technol.*, 2009.

<https://doi.org/10.1007/s11666-009-9314-5>.

- [71] M. Prystay, P. Gougeon, C. Moreau, Correlation between Particle Temperature and Velocity and the Structure of Plasma Sprayed Zirconia Coatings, in: *Therm. Spray 1996 Proc. from Natl. Therm. Spray Conf.*, 1996. <https://doi.org/10.31399/asm.cp.itsc1996p0517>.
- [72] M. Vardelle, A. Vardelle, A.C. Leger, P. Fauchais, D. Gobin, Influence of particle parameters at impact on splat formation and solidification in plasma spraying processes, *J. Therm. Spray Technol.* (1995). <https://doi.org/10.1007/BF02648528>.
- [73] J.G. Legoux, B. Arsenault, V. Bouyer, C. Moreau, L. Leblanc, Evaluation of four high velocity thermal spray guns using WC-10% CO-4% Cr cermets, *J. Therm. Spray Technol.* (2002). <https://doi.org/10.1361/105996302770349014>.
- [74] F. Venturi, T. Hussain, Radial Injection in Suspension High Velocity Oxy-Fuel (S-HVOF) Thermal Spray of Graphene Nanoplatelets for Tribology, *J. Therm. Spray Technol.* (2020). <https://doi.org/10.1007/s11666-019-00957-y>.
- [75] P. Wang, S.C.M. Yu, H.W. Ng, Correlation of plasma sprayed coating deposition efficiency with volume flux measurements by Phase Doppler Anemometry (PDA), *Plasma Chem. Plasma Process.* (2007). <https://doi.org/10.1007/s11090-007-9065-3>.
- [76] N.H. Faisal, R. Ahmed, R.L. Reuben, B. Allcock, AE monitoring and analysis of HVOF thermal spraying process, *J. Therm. Spray Technol.* (2011). <https://doi.org/10.1007/s11666-011-9620-6>.
- [77] C.K. Lin, C.C. Berndt, Acoustic emission studies on thermal spray materials, *Surf. Coatings Technol.* (1998). [https://doi.org/10.1016/S0257-8972\(97\)00531-8](https://doi.org/10.1016/S0257-8972(97)00531-8).
- [78] X.Q. Ma, S. Cho, M. Takemoto, Acoustic emission source analysis of plasma sprayed thermal barrier coating during four-point bend tests, *Surf. Coatings Technol.* (2001). [https://doi.org/10.1016/S0257-8972\(01\)00995-1](https://doi.org/10.1016/S0257-8972(01)00995-1).
- [79] W. Tillmann, F. Walther, W. Luo, M. Haack, J. Nellesen, M. Knyazeva, In Situ Acoustic Monitoring of Thermal Spray Process Using High-Frequency Impulse Measurements, *J. Therm. Spray Technol.* (2018). <https://doi.org/10.1007/s11666-017-0673-z>.
- [80] S. Nishinoiri, M. Enoki, K. Tomita, In situ monitoring of microfracture during plasma spray coating by laser AE technique, *Sci. Technol. Adv. Mater.* (2003). <https://doi.org/10.1016/j.stam.2003.12.011>.
- [81] Y. Wang, P. Zhao, Noncontact acoustic analysis monitoring of plasma arc welding, *Int. J. Press. Vessel. Pip.* (2001). [https://doi.org/10.1016/S0308-0161\(00\)00085-5](https://doi.org/10.1016/S0308-0161(00)00085-5).
- [82] W. Huang, R. Kovacevic, Feasibility study of using acoustic signals for online monitoring of the depth of weld in the laser welding of high-strength steels, *Proc. Inst. Mech. Eng. Part B J. Eng. Manuf.* (2009). <https://doi.org/10.1243/09544054JEM1320>.
- [83] E. Saad, H. Wang, R. Kovacevic, Classification of molten pool modes in variable polarity plasma arc welding based on acoustic signature, *J. Mater. Process. Technol.* (2006). <https://doi.org/10.1016/j.jmatprotec.2005.03.020>.
- [84] L. Grad, J. Grum, I. Polajnar, J.M. Slabe, Feasibility study of acoustic signals for on-line monitoring in short circuit gas metal arc welding, *Int. J. Mach. Tools Manuf.*

- (2004). <https://doi.org/10.1016/j.ijmachtools.2003.10.016>.
- [85] Y.W. Wang, Q. Chen, Z.G. Sun, J.W. Sun, Relationship between sound signal and weld pool status in plasma arc welding, *Trans. Nonferrous Met. Soc. China (English Ed.)* (2001).
 - [86] C. Pierlot, L. Pawlowski, M. Bigan, P. Chagnon, Design of experiments in thermal spraying: A review, *Surf. Coatings Technol.* (2008). <https://doi.org/10.1016/j.surfcoat.2008.04.031>.
 - [87] R.A. Miller, Thermal spray: International advances in coatings technology, *J. Therm. Spray Technol.* (1992). <https://doi.org/10.1007/BF02647153>.
 - [88] M. Ebrahimi, F. Mahboubi, M.R. Naimi-Jamal, RSM base study of the effect of deposition temperature and hydrogen flow on the wear behavior of DLC films, *Tribol. Int.* (2015). <https://doi.org/10.1016/j.triboint.2015.06.026>.
 - [89] R. Shankar, K.R. Balasubramanian, S.P. Sivapirakasam, K. Ravikumar, ANN and RSM models approach for optimization of HVOF coating, *Mater. Today Proc.* (2020). <https://doi.org/10.1016/j.matpr.2020.01.211>.
 - [90] A. Iwaniak, Ł. Norymberczyk, The effect of selected laser beam micromilling parameters on the surface layer structure of High Velocity Oxygen Fuel sprayed WC-CoCr coating, *Surf. Coatings Technol.* (2019). <https://doi.org/10.1016/j.surfcoat.2019.125043>.
 - [91] B.T. Lin, M. Der Jean, J.H. Chou, Using response surface methodology for optimizing deposited partially stabilized zirconia in plasma spraying, *Appl. Surf. Sci.* (2007). <https://doi.org/10.1016/j.apsusc.2006.07.021>.
 - [92] J. Cizek, K.A. Khor, Z. Prochazka, Influence of spraying conditions on thermal and velocity properties of plasma sprayed hydroxyapatite, *Mater. Sci. Eng. C.* (2007). <https://doi.org/10.1016/j.msec.2006.05.002>.
 - [93] J.F. Li, H. Liao, B. Normand, C. Cordier, G. Maurin, J. Foct, C. Coddet, Uniform design method for optimization of process parameters of plasma sprayed TiN coatings, *Surf. Coatings Technol.* (2003). [https://doi.org/10.1016/S0257-8972\(03\)00019-7](https://doi.org/10.1016/S0257-8972(03)00019-7).
 - [94] M. Atiq Ur Rehman, F.E. Bastan, B. Haider, A.R. Boccaccini, Electrophoretic deposition of PEEK/bioactive glass composite coatings for orthopedic implants: A design of experiments (DoE) study, *Mater. Des.* (2017). <https://doi.org/10.1016/j.matdes.2017.05.045>.
 - [95] G.G. Vinokurov, O.N. Popov, Statistical approaches for the description formations and wear of the macrostructure of powder coatings and materials, in: *Procedia Struct. Integr.*, 2019. <https://doi.org/10.1016/j.prostr.2019.12.150>.
 - [96] S. Özel, E. Vural, M. Binici, Optimization of the effect of thermal barrier coating (TBC) on diesel engine performance by Taguchi method, *Fuel*. (2020). <https://doi.org/10.1016/j.fuel.2019.116537>.
 - [97] A. van Kampen, R. Kohlus, Statistical modelling of coating layer thickness distributions: Influence of overspray on coating quality, *Powder Technol.* (2018). <https://doi.org/10.1016/j.powtec.2017.11.031>.
 - [98] B. Yegnanarayana, Artificial neural networks for pattern recognition, *Sadhana*. (1994).

<https://doi.org/10.1007/BF02811896>.

- [99] J. Teuwen, N. Moriaikov, Convolutional neural networks, in: *Handb. Med. Image Comput. Comput. Assist. Interv.*, 2019. <https://doi.org/10.1016/B978-0-12-816176-0.00025-9>.
- [100] S.A. Marhon, C.J.F. Cameron, S.C. Kremer, Recurrent Neural Networks, *Intell. Syst. Ref. Libr.* (2013). https://doi.org/10.1007/978-3-642-36657-4_2.
- [101] Y. Lexuan, H. No, Improvements on Activation Functions in ANN: An Overview, *Manag. Sci. Eng.* (2020).
- [102] T. Poggio, Q. Liao, A. Banburski, Complexity control by gradient descent in deep networks, *Nat. Commun.* (2020). <https://doi.org/10.1038/s41467-020-14663-9>.
- [103] S. Bock, M. Weis, A Proof of Local Convergence for the Adam Optimizer, in: *Proc. Int. Jt. Conf. Neural Networks*, 2019. <https://doi.org/10.1109/IJCNN.2019.8852239>.
- [104] Y. Liu, Y. Gao, W. Yin, An improved analysis of stochastic gradient descent with momentum, in: *Adv. Neural Inf. Process. Syst.*, 2020.
- [105] D. Vijendra Babu, C. Karthikeyan, Shreya, A. Kumar, Performance analysis of cost and accuracy for whale swarm and RMSprop optimizer, in: *IOP Conf. Ser. Mater. Sci. Eng.*, 2020. <https://doi.org/10.1088/1757-899X/993/1/012080>.
- [106] Simple Introduction to Convolutional Neural Networks | by Matthew Stewart, PhD Researcher | Towards Data Science, (n.d.). <https://towardsdatascience.com/simple-introduction-to-convolutional-neural-networks-cdf8d3077bac> (accessed September 6, 2021).
- [107] J. Zhu, X. Wang, L. Kou, L. Zheng, H. Zhang, Prediction of control parameters corresponding to in-flight particles in atmospheric plasma spray employing convolutional neural networks, *Surf. Coatings Technol.* (2020). <https://doi.org/10.1016/j.surfcoat.2020.125862>.
- [108] S. Guessasma, G. Montavon, C. Coddet, On the implementation of neural network concept to optimize thermal spray deposition process, in: *Mater. Res. Soc. Symp. - Proc.*, 2002. <https://doi.org/10.1557/proc-700-s8.2>.
- [109] S. Guessasma, G. Montavon, C. Coddet, Neural computation to predict in-flight particle characteristic dependences from processing parameters in the APS process, *J. Therm. Spray Technol.* (2004). <https://doi.org/10.1361/10599630419391>.
- [110] S. Guessasma, G. Montavon, C. Coddet, Analysis of the influence of atmospheric plasma spray (APS) parameters on adhesion properties of alumina-titania coatings, *J. Adhes. Sci. Technol.* (2004). <https://doi.org/10.1163/156856104323016388>.
- [111] S. Guessasma, Z. Salhi, G. Montavon, P. Gougeon, C. Coddet, Artificial intelligence implementation in the APS process diagnostic, *Mater. Sci. Eng. B Solid-State Mater. Adv. Technol.* (2004). <https://doi.org/10.1016/j.mseb.2004.03.017>.
- [112] T.A. Choudhury, N. Hosseinzadeh, C.C. Berndt, Artificial Neural Network application for predicting in-flight particle characteristics of an atmospheric plasma spray process, *Surf. Coatings Technol.* (2011). <https://doi.org/10.1016/j.surfcoat.2011.04.099>.
- [113] T.A. Choudhury, N. Hosseinzadeh, C.C. Berndt, Improving the generalization ability

- of an artificial neural network in predicting in-flight particle characteristics of an atmospheric plasma spray process, *J. Therm. Spray Technol.* (2012). <https://doi.org/10.1007/s11666-012-9775-9>.
- [114] T. Liu, M.P. Planche, A.F. Kanta, S. Deng, G. Montavon, K. Deng, Z.M. Ren, Plasma spray process operating parameters optimization based on artificial intelligence, *Plasma Chem. Plasma Process.* (2013). <https://doi.org/10.1007/s11090-013-9475-3>.
 - [115] C.M. Lin, S.H. Yen, C.Y. Su, Measurement and optimization of atmospheric plasma sprayed CoMoCrSi coatings parameters on Ti-6Al-4V substrates affecting microstructural and properties using hybrid abductor induction mechanism, *Meas. J. Int. Meas. Confed.* (2016). <https://doi.org/10.1016/j.measurement.2016.07.077>.
 - [116] T.A. Choudhury, C.C. Berndt, Z. Man, Modular implementation of artificial neural network in predicting in-flight particle characteristics of an atmospheric plasma spray process, *Eng. Appl. Artif. Intell.* (2015). <https://doi.org/10.1016/j.engappai.2015.06.015>.
 - [117] M. Liu, Z. Yu, H. Wu, S. Deng, Y. Zhang, H. Liao, Artificial intelligent aided analysis and prediction of high velocity oxy fuel (HVOF) sprayed Cr₃C₂-25NiCr coatings, in: *Proc. Int. Therm. Spray Conf.*, 2019.
 - [118] M. Liu, Z. Yu, Y. Zhang, H. Wu, H. Liao, S. Deng, Prediction and analysis of high velocity oxy fuel (HVOF) sprayed coating using artificial neural network, *Surf. Coatings Technol.* (2019). <https://doi.org/10.1016/j.surfcoat.2019.124988>.
 - [119] J. Singh, Analysis on suitability of HVOF sprayed Ni-20Al, Ni-20Cr and Al-20Ti coatings in coal-ash slurry conditions using artificial neural network model, *Ind. Lubr. Tribol.* (2019). <https://doi.org/10.1108/ILT-12-2018-0460>.
 - [120] M. Liu, Z. Yu, H. Wu, H. Liao, Q. Zhu, S. Deng, Implementation of Artificial Neural Networks for Forecasting the HVOF Spray Process and HVOF Sprayed Coatings, *J. Therm. Spray Technol.* (2021). <https://doi.org/10.1007/s11666-021-01213-y>.
 - [121] H. Cetinel, The artificial neural network based prediction of friction properties of Al₂O₃-TiO₂ coatings, *Ind. Lubr. Tribol.* (2012). <https://doi.org/10.1108/00368791211249674>.
 - [122] S. Kamnis, K. Malamousi, A. Marrs, B. Allcock, K. Delibasis, Aeroacoustics and Artificial Neural Network Modeling of Airborne Acoustic Emissions During High Kinetic Energy Thermal Spraying, *J. Therm. Spray Technol.* (2019). <https://doi.org/10.1007/s11666-019-00874-0>.
 - [123] J. Salamon, J.P. Bello, Deep Convolutional Neural Networks and Data Augmentation for Environmental Sound Classification, *IEEE Signal Process. Lett.* (2017). <https://doi.org/10.1109/LSP.2017.2657381>.
 - [124] N. Takahashi, M. Gygli, L. Van Gool, AENet: Learning Deep Audio Features for Video Analysis, *IEEE Trans. Multimed.* (2018). <https://doi.org/10.1109/TMM.2017.2751969>.
 - [125] A. Pandey, D. Wang, A New Framework for CNN-Based Speech Enhancement in the Time Domain, *IEEE/ACM Trans. Audio Speech Lang. Process.* (2019). <https://doi.org/10.1109/TASLP.2019.2913512>.
 - [126] X. Zhai, C. Tin, Automated ECG Classification Using Dual Heartbeat Coupling Based

- on Convolutional Neural Network, IEEE Access. (2018).
<https://doi.org/10.1109/ACCESS.2018.2833841>.
- [127] A. Kothuru, S.P. Nooka, R. Liu, Audio-Based Tool Condition Monitoring in Milling of the Workpiece Material With the Hardness Variation Using Support Vector Machines and Convolutional Neural Networks, J. Manuf. Sci. Eng. Trans. ASME. (2018).
<https://doi.org/10.1115/1.4040874>.
 - [128] W. Zhang, W. Lei, X. Xu, X. Xing, Improved music genre classification with convolutional neural networks, in: Proc. Annu. Conf. Int. Speech Commun. Assoc. INTERSPEECH, 2016. <https://doi.org/10.21437/Interspeech.2016-1236>.
 - [129] F. Demir, D.A. Abdullah, A. Sengur, A New Deep CNN Model for Environmental Sound Classification, IEEE Access. (2020).
<https://doi.org/10.1109/ACCESS.2020.2984903>.
 - [130] Y. Tokozume, T. Harada, Learning environmental sounds with end-to-end convolutional neural network, in: ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc., 2017. <https://doi.org/10.1109/ICASSP.2017.7952651>.
 - [131] K. Choi, G. Fazekas, M. Sandler, K. Cho, Convolutional recurrent neural networks for music classification, in: ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc., 2017. <https://doi.org/10.1109/ICASSP.2017.7952585>.
 - [132] Y. Banadaki, N. Razaviarab, H. Fekrmandi, S. Sharifi, Toward enabling a reliable quality monitoring system for additive manufacturing process using deep convolutional neural networks, ArXiv. (2020).
 - [133] Q. Wang, W. Jiao, P. Wang, Y.M. Zhang, A tutorial on deep learning-based data analytics in manufacturing through a welding case study, J. Manuf. Process. (2020).
<https://doi.org/10.1016/j.jmapro.2020.04.044>.
 - [134] Z. Zhang, B. Li, W. Zhang, R. Lu, S. Wada, Y. Zhang, Real-time penetration state monitoring using convolutional neural network for laser welding of tailor rolled blanks, J. Manuf. Syst. (2020). <https://doi.org/10.1016/j.jmsy.2020.01.006>.
 - [135] A.P. Garland, B.C. White, B.H. Jared, M. Heiden, E. Donahue, B.L. Boyce, Deep Convolutional Neural Networks as a Rapid Screening Tool for Complex Additively Manufactured Structures, Addit. Manuf. (2020).
<https://doi.org/10.1016/j.addma.2020.101217>.
 - [136] B. Zhang, P. Jaiswal, R. Rai, P. Guerrier, G. Baggs, Convolutional neural network-based inspection of metal additive manufacturing parts, Rapid Prototyp. J. (2019).
<https://doi.org/10.1108/RPJ-04-2018-0096>.
 - [137] Y. Zhang, G.S. Hong, D. Ye, K. Zhu, J.Y.H. Fuh, Extraction and evaluation of melt pool, plume and spatter information for powder-bed fusion AM process monitoring, Mater. Des. (2018). <https://doi.org/10.1016/j.matdes.2018.07.002>.
 - [138] J. Wigren, K. Täng, Quality considerations for the evaluation of thermal spray coatings, J. Therm. Spray Technol. (2007). <https://doi.org/10.1007/s11666-007-9054-3>.
 - [139] ASTM E1920-03(2014), Standard Guide for Metallographic Preparation of Thermal Sprayed Coatings, ASTM International, West Conshohocken, PA, 2014, www.astm.org, (n.d.). <https://doi.org/10.1520/E1920-03R14>.

- [140] ASTM E3-11(2017), Standard Guide for Preparation of Metallographic Specimens, ASTM International, West Conshohocken, PA, DOI: 10.1520/E0003-11R17 2017, www.astm.org, (n.d.).
- [141] A. Kordijazi, T. Zhao, J. Zhang, K. Alrfou, P. Rohatgi, A Review of Application of Machine Learning in Design, Synthesis, and Characterization of Metal Matrix Composites: Current Status and Emerging Applications, JOM. (2021). <https://doi.org/10.1007/s11837-021-04701-2>.
- [142] E.A. Holm, R. Cohn, N. Gao, A.R. Kitahara, T.P. Matson, B. Lei, S.R. Yarasi, Overview: Computer Vision and Machine Learning for Microstructural Characterization and Analysis, Metall. Mater. Trans. A Phys. Metall. Mater. Sci. (2020). <https://doi.org/10.1007/s11661-020-06008-4>.
- [143] O.B. Abouelatta, Classification of copper alloys microstructure using image processing and neural network, J. Am. Sci. (2013).
- [144] A. Chowdhury, E. Kautz, B. Yener, D. Lewis, Image driven machine learning methods for microstructure recognition, Comput. Mater. Sci. (2016). <https://doi.org/10.1016/j.commatsci.2016.05.034>.
- [145] V.H.C. De Albuquerque, P.C. Cortez, A.R. De Alexandria, J.M.R.S. Tavares, A new solution for automatic microstructures analysis from images based on a backpropagation artificial neural network, Nondestruct. Test. Eval. (2008). <https://doi.org/10.1080/10589750802258986>.
- [146] Z.L. Wang, T. Ogawa, Y. Adachi, Machine-Learning-Based Image Similarity Analysis for Use in Materials Characterization, Adv. Theory Simulations. (2020). <https://doi.org/10.1002/adts.201900237>.
- [147] B. Girod, Psychovisual Aspects Of Image Processing: What's Wrong With Mean Squared Error?, in: 2005. <https://doi.org/10.1109/mdsp.1991.639240>.
- [148] Z. Wang, A.C. Bovik, A universal image quality index, IEEE Signal Process. Lett. (2002). <https://doi.org/10.1109/97.995823>.
- [149] Z. Wang, A.C. Bovik, L. Lu, Why is image quality assessment so difficult?, ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc. (2002). <https://doi.org/10.1109/ICASSP.2002.5745362>.
- [150] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: From error visibility to structural similarity, IEEE Trans. Image Process. (2004). <https://doi.org/10.1109/TIP.2003.819861>.
- [151] A.N. Avanaki, Exact global histogram specification optimized for structural similarity, Opt. Rev. (2009). <https://doi.org/10.1007/s10043-009-0119-z>.
- [152] F. Sun, D.Q. Zhang, L. Cheng, P. Zheng, D.M. Liao, B. Zhu, Microstructure Evolution Modeling and Simulation for Dynamic Recrystallization of Cr12MoV Die Steel During Hot Compression Based on Real Metallographic Image, Met. Mater. Int. (2019). <https://doi.org/10.1007/s12540-019-00249-8>.
- [153] S.K. Basantia, A. Bhattacharya, N. Khutia, D. Das, Plastic Behavior of Ferrite–Pearlite, Ferrite–Bainite and Ferrite–Martensite Steels: Experiments and Micromechanical Modelling, Met. Mater. Int. (2021). <https://doi.org/10.1007/s12540-019-00519-5>.

- [154] H. Toda, A. Takijiri, M. Azuma, S. Yabu, K. Hayashi, D. Seo, M. Kobayashi, K. Hirayama, A. Takeuchi, K. Uesugi, Damage micromechanisms in dual-phase steel investigated with combined phase- and absorption-contrast tomography, *Acta Mater.* (2017). <https://doi.org/10.1016/j.actamat.2017.01.010>.
- [155] S. Nickerson, Y. Shu, D. Zhong, C. Könke, A. Tandia, Permeability of porous ceramics by X-ray CT image analysis, *Acta Mater.* (2019). <https://doi.org/10.1016/j.actamat.2019.04.053>.
- [156] V. Katranidis, S. Gu, D.C. Cox, M.J. Whiting, S. Kamnis, FIB-SEM Sectioning Study of Decarburization Products in the Microstructure of HVOF-Sprayed WC-Co Coatings, *J. Therm. Spray Technol.* 27 (2018). <https://doi.org/10.1007/s11666-018-0721-3>.
- [157] V. Katranidis, S. Kamnis, B. Allcock, S. Gu, Effects and Interplays of Spray Angle and Stand-off Distance on the Sliding Wear Behavior of HVOF WC-17Co Coatings, *J. Therm. Spray Technol.* 28 (2019). <https://doi.org/10.1007/s11666-019-00831-x>.
- [158] C. Bao, C. Ji, H.F. Poulsen, M. Li, Missing information and data fidelity in digital microstructure acquisition, *Acta Mater.* (2019). <https://doi.org/10.1016/j.actamat.2019.05.012>.
- [159] M. Baniassadi, H. Garmestani, D.S. Li, S. Ahzi, M. Khaleel, X. Sun, Three-phase solid oxide fuel cell anode microstructure realization using two-point correlation functions, *Acta Mater.* (2011). <https://doi.org/10.1016/j.actamat.2010.08.012>.
- [160] V. Sundararaghavan, Reconstruction of three-dimensional anisotropic microstructures from two-dimensional micrographs imaged on orthogonal planes, *Integr. Mater. Manuf. Innov.* (2014). <https://doi.org/10.1186/s40192-014-0019-3>.
- [161] Y. Zhang, M. Yan, Y. Wan, Z. Jiao, Y. Chen, F. Chen, C. Xia, M. Ni, High-throughput 3D reconstruction of stochastic heterogeneous microstructures in energy storage materials, *Npj Comput. Mater.* (2019). <https://doi.org/10.1038/s41524-019-0149-4>.
- [162] C. Dong, C.C. Loy, K. He, X. Tang, Image Super-Resolution Using Deep Convolutional Networks, *IEEE Trans. Pattern Anal. Mach. Intell.* (2016). <https://doi.org/10.1109/TPAMI.2015.2439281>.
- [163] K. Shiina, H. Mori, Y. Tomita, H.K. Lee, Y. Okabe, Inverse renormalization group based on image super-resolution using deep convolutional networks, *Sci. Rep.* (2021). <https://doi.org/10.1038/s41598-021-88605-w>.
- [164] B. Lim, S. Son, H. Kim, S. Nah, K.M. Lee, Enhanced Deep Residual Networks for Single Image Super-Resolution, in: *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, 2017. <https://doi.org/10.1109/CVPRW.2017.151>.
- [165] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, W. Shi, Photo-realistic single image super-resolution using a generative adversarial network, in: *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, 2017. <https://doi.org/10.1109/CVPR.2017.19>.
- [166] D. Kim, M. Kim, G. Kwon, D.S. Kim, Progressive face super-resolution via attention to facial landmark, in: *30th Br. Mach. Vis. Conf. 2019, BMVC 2019*, 2020.
- [167] C. Dong, C.C. Loy, X. Tang, Accelerating the super-resolution convolutional neural network, in: *Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell.*

- Lect. Notes Bioinformatics), 2016. https://doi.org/10.1007/978-3-319-46475-6_25.
- [168] W.S. Lai, J. Bin Huang, N. Ahuja, M.H. Yang, Fast and Accurate Image Super-Resolution with Deep Laplacian Pyramid Networks, *IEEE Trans. Pattern Anal. Mach. Intell.* (2019). <https://doi.org/10.1109/TPAMI.2018.2865304>.
 - [169] S. Ayas, M. Ekinici, Microscopic image super resolution using deep convolutional neural networks, *Multimed. Tools Appl.* (2020). <https://doi.org/10.1007/s11042-019-7397-7>.
 - [170] K. Hagita, T. Higuchi, H. Jinnai, Super-resolution for asymmetric resolution of FIB-SEM 3D imaging using AI with deep learning, *Sci. Rep.* (2018). <https://doi.org/10.1038/s41598-018-24330-1>.
 - [171] K. de Haan, Z.S. Ballard, Y. Rivenson, Y. Wu, A. Ozcan, Resolution enhancement in scanning electron microscopy using deep learning, *Sci. Rep.* (2019). <https://doi.org/10.1038/s41598-019-48444-2>.
 - [172] J. Jung, J. Na, H.K. Park, J.M. Park, G. Kim, S. Lee, H.S. Kim, Super-resolving material microstructure image via deep learning for microstructure characterization and mechanical behavior analysis, *Npj Comput. Mater.* (2021). <https://doi.org/10.1038/s41524-021-00568-8>.
 - [173] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2016. <https://doi.org/10.1109/CVPR.2016.90>.
 - [174] W. Shi, J. Caballero, F. Huszar, J. Totz, A.P. Aitken, R. Bishop, D. Rueckert, Z. Wang, Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network, in: *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2016. <https://doi.org/10.1109/CVPR.2016.207>.
 - [175] U. Handalage, L. Kuganandamurthy, Real-Time Object Detection Using YOLO: A Review, *ResearchGate*. (2021).
 - [176] M. Wang, P. Moin, Dynamic wall modeling for large-eddy simulation of complex turbulent flows, *Phys. Fluids*. (2002). <https://doi.org/10.1063/1.1476668>.
 - [177] A. Uzun, A.S. Lyrantzis, G.A. Blaisdell, Coupling of Integral Acoustics Methods with LES for Jet Noise Prediction, *Int. J. Aeroacoustics*. (2004). <https://doi.org/10.1260/1475472043499290>.
 - [178] K. Malamousi, K. Delibasis, B. Allcock, S. Kamnis, Digital transformation of thermal and cold spray processes with emphasis on machine learning, *Surf. Coatings Technol.* (2022). <https://doi.org/10.1016/j.surfcoat.2022.128138>.

Appendix A: Audio Classification Using CNNs

Implementation code:

Implementation Code Link:

<https://colab.research.google.com/drive/176qJoSRgjEVWhr6kgGklDFGYgm0ugavd?usp=sharing>

```

import os
import pathlib

import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
import tensorflow as tf

from tensorflow.keras.layers.experimental import preprocessing
from tensorflow.keras import layers
from tensorflow.keras import models
from IPython import display

# Set seed for experiment reproducibility
seed = 42
tf.random.set_seed(seed)
np.random.seed(seed)

data_dir = pathlib.Path('')
if not data_dir.exists():
    tf.keras.utils.get_file(
        '.zip',
        origin="http://",
        extract=True,
        cache_dir='.', cache_subdir='')

commands = np.array(tf.io.gfile.listdir(str(data_dir)))
commands = commands[commands != 'README.md']
print('Commands:', commands)

Commands: ['High_Powder' 'Low_Powder' 'Medium_Powder' 'No_Powder']

filenames = tf.io.gfile.glob(str(data_dir) + '/*/*')
filenames = tf.random.shuffle(filenames)
num_samples = len(filenames)
print('Number of total examples:', num_samples)
print('Number of examples per label:',
      len(tf.io.gfile.listdir(str(data_dir/commands[0]))))
print('Example file tensor:', filenames[0])

Number of total examples: 684
Number of examples per label: 171
Example file tensor: tf.Tensor(b'simpleaudiohvf\condition\No_Powder\91_no_powder.wav', shape=(), dtype=string)

train_files = filenames[: ]
val_files = filenames[584: 584 + 50]
test_files = filenames[-50:]

print('Training set size', len(train_files))
print('Validation set size', len(val_files))
print('Test set size', len(test_files))

Training set size 684
Validation set size 50
Test set size 50

def decode_audio(audio_binary):
    audio, _ = tf.audio.decode_wav(audio_binary)
    return tf.squeeze(audio, axis=-1)

def get_label(file_path):
    parts = tf.strings.split(file_path, os.path.sep)

    # Note: You'll use indexing here instead of tuple unpacking to enable this
    # to work in a TensorFlow graph.
    return parts[-2]

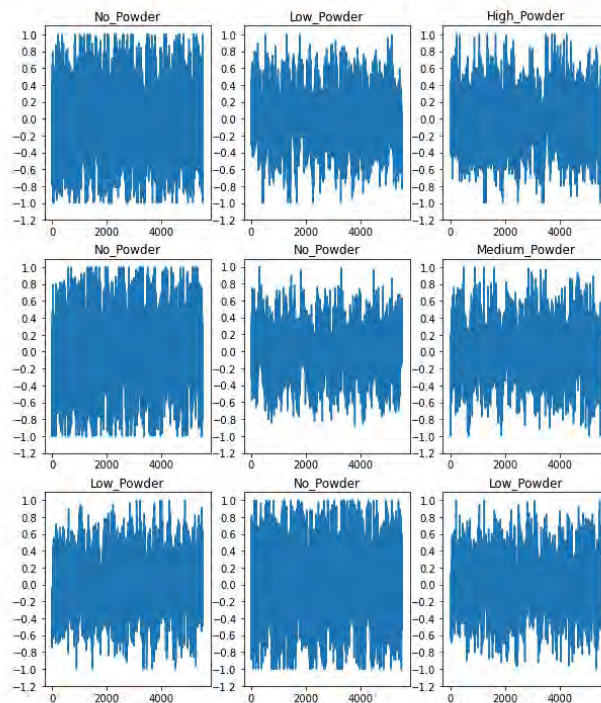
```

```
def get_waveform_and_label(file_path):
    label = get_label(file_path)
    audio_binary = tf.io.read_file(file_path)
    waveform = decode_audio(audio_binary)
    return waveform, label

AUTOTUNE = tf.data.experimental.AUTOTUNE
files_ds = tf.data.Dataset.from_tensor_slices(train_files)
waveform_ds = files_ds.map(get_waveform_and_label, num_parallel_calls=AUTOTUNE)
```

```
rows = 3
cols = 3
n = rows*cols
fig, axes = plt.subplots(rows, cols, figsize=(10, 12))
for i, (audio, label) in enumerate(waveform_ds.take(n)):
    r = i // cols
    c = i % cols
    ax = axes[r][c]
    ax.plot(audio.numpy())
    ax.set_yticks(np.arange(-1.2, 1.2, 0.2))
    label = label.numpy().decode('utf-8')
    ax.set_title(label)

plt.show()
```



```
def get_spectrogram(waveform):
    # Padding for files with less than 16000 samples
    zero_padding = tf.zeros([16000] - tf.shape(waveform), dtype=tf.float32)

    # Concatenate audio with padding so that all audio clips will be of the
    # same length
    waveform = tf.cast(waveform, tf.float32)
    equal_length = tf.concat([waveform, zero_padding], 0)
    spectrogram = tf.signal.stft(
        equal_length, frame_length=255, frame_step=128)

    spectrogram = tf.abs(spectrogram)

    return spectrogram
```

```

for waveform, label in waveform_ds.take(1):
    label = label.numpy().decode('utf-8')
    spectrogram = get_spectrogram(waveform)

print('Label:', label)
print('Waveform shape:', waveform.shape)
print('Spectrogram shape:', spectrogram.shape)
print('Audio playback')
display.display(display.Audio(waveform, rate=16000))

    Label: No_Powder
    Waveform shape: (5513,)
    Spectrogram shape: (124, 129)
    Audio playback

    0:00 / 0:00

def plot_spectrogram(spectrogram, ax):
    # Convert to frequencies to log scale and transpose so that the time is
    # represented in the x-axis (columns).
    log_spec = np.log(spectrogram.T)
    height = log_spec.shape[0]
    X = np.arange(16000, step=height + 1)
    Y = range(height)
    ax.pcolormesh(X, Y, log_spec)

fig, axes = plt.subplots(2, figsize=(12, 8))
timescale = np.arange(waveform.shape[0])
axes[0].plot(timescale, waveform.numpy())
axes[0].set_title('Waveform')
axes[0].set_xlim([0, 16000])
plot_spectrogram(spectrogram.numpy(), axes[1])
axes[1].set_title('Spectrogram')
plt.show()

def get_spectrogram_and_label_id(audio, label):
    spectrogram = get_spectrogram(audio)
    spectrogram = tf.expand_dims(spectrogram, -1)
    label_id = tf.argmax(label == commands)
    return spectrogram, label_id

spectrogram_ds = waveform_ds.map(
    get_spectrogram_and_label_id, num_parallel_calls=AUTOTUNE)

rows = 3
cols = 3
n = rows*cols
fig, axes = plt.subplots(rows, cols, figsize=(10, 10))
for i, (spectrogram, label_id) in enumerate(spectrogram_ds.take(n)):
    r = i // cols
    c = i % cols
    ax = axes[r][c]
    plot_spectrogram(np.squeeze(spectrogram.numpy()), ax)
    ax.set_title(commands[label_id.numpy()])
    ax.axis('off')

plt.show()

def preprocess_dataset(files):
    files_ds = tf.data.Dataset.from_tensor_slices(files)
    output_ds = files_ds.map(get_waveform_and_label, num_parallel_calls=AUTOTUNE)
    output_ds = output_ds.map(
        get_spectrogram_and_label_id, num_parallel_calls=AUTOTUNE)
    return output_ds

train_ds = spectrogram_ds
val_ds = preprocess_dataset(val_files)
test_ds = preprocess_dataset(test_files)

```

```

batch_size = 64
train_ds = train_ds.batch(batch_size)
val_ds = val_ds.batch(batch_size)

train_ds = train_ds.cache().prefetch(AUTOTUNE)
val_ds = val_ds.cache().prefetch(AUTOTUNE)

for spectrogram, _ in spectrogram_ds.take(1):
    input_shape = spectrogram.shape
    print('Input shape:', input_shape)
    num_labels = len(commands)

norm_layer = preprocessing.Normalization()
norm_layer.adapt(spectrogram_ds.map(lambda x, _: x))

model = models.Sequential([
    layers.Input(shape=input_shape),
    preprocessing.Resizing(32, 32),
    norm_layer,
    layers.Conv2D(32, 3, activation='relu'),
    layers.Conv2D(64, 3, activation='relu'),
    layers.MaxPooling2D(),
    layers.Dropout(0.25),
    layers.Flatten(),
    layers.Dense(128, activation='relu'),
    layers.Dropout(0.5),
    layers.Dense(num_labels),
])

model.summary()

Input shape: (124, 129, 1)
Model: "sequential"

```

Layer (type)	Output Shape	Param #
resizing (Resizing)	(None, 32, 32, 1)	0
normalization (Normalization)	(None, 32, 32, 1)	3
conv2d (Conv2D)	(None, 30, 30, 32)	320
conv2d_1 (Conv2D)	(None, 28, 28, 64)	18496
max_pooling2d (MaxPooling2D)	(None, 14, 14, 64)	0
dropout (Dropout)	(None, 14, 14, 64)	0
flatten (Flatten)	(None, 12544)	0
dense (Dense)	(None, 128)	1605760
dropout_1 (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 4)	516

```

Total params: 1,625,095
Trainable params: 1,625,092
Non-trainable params: 3

model.compile(
    optimizer=tf.keras.optimizers.Adam(),
    loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
    metrics=['accuracy'],
)

EPOCHS = 50
history = model.fit(
    train_ds,
    validation_data=val_ds,
    epochs=EPOCHS,
    callbacks=tf.keras.callbacks.EarlyStopping(verbose=1, patience=2),
)

```

```

metrics = history.history
plt.plot(history.epoch, metrics['loss'], metrics['val_loss'])
plt.legend(['loss', 'val_loss'])
plt.show()

test_audio = []
test_labels = []

for audio, label in test_ds:
    test_audio.append(audio.numpy())
    test_labels.append(label.numpy())

test_audio = np.array(test_audio)
test_labels = np.array(test_labels)

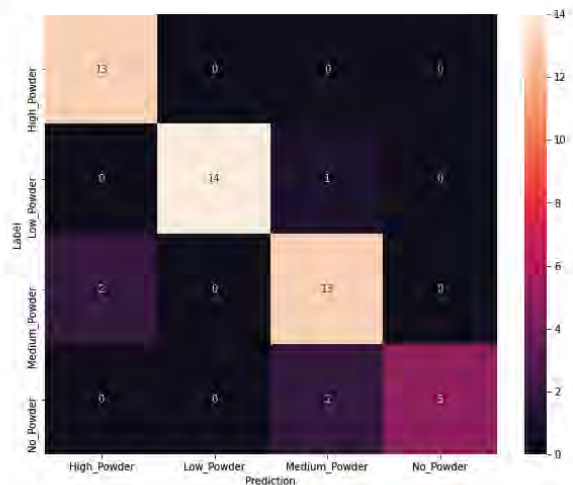
y_pred = np.argmax(model.predict(test_audio), axis=1)
y_true = test_labels

test_acc = sum(y_pred == y_true) / len(y_true)
print(f'Test set accuracy: {test_acc:.8%}')

Test set accuracy: 98%

confusion_mtx = tf.math.confusion_matrix(y_true, y_pred)
plt.figure(figsize=(10, 8))
sns.heatmap(confusion_mtx, xticklabels=commands, yticklabels=commands,
            annot=True, fmt='g')
plt.xlabel('Prediction')
plt.ylabel('Label')
plt.show()

```



```

sample_file = data_dir/''

sample_ds = preprocess_dataset([str(sample_file)])

for spectrogram, label in sample_ds.batch(1):
    prediction = model(spectrogram)
    plt.bar(commands, tf.nn.softmax(prediction[0]))
    plt.title(f'Predictions for "{commands[label[0]]}"')
    plt.show()

```

Appendix B: Spray Process Image Classification Using CNNs

Implementation Code Link:

<https://colab.research.google.com/drive/1sAYct7sU7ifuWepGIGVvxqxOKVd23g9v?usp=sharing>

```

from keras.preprocessing.image import ImageDataGenerator, load_img, img_to_array
from keras.models import Sequential
from keras.layers import Conv2D, MaxPooling2D
from keras.layers import Activation, Dropout, Flatten, Dense
from keras import backend as K
import matplotlib.pyplot as plt
from keras.models import model_from_json
from keras.models import load_model
import os
import cv2
import PIL
import numpy as np
import pandas as pd
import datetime
import tensorflow as tf

```

```

# dimensions of our images.
img_width, img_height = 50, 50
train_data_dir = ''
validation_data_dir = ''
nb_train_samples = 2800
nb_validation_samples = 290
epochs = 8
batch_size = 200

```

```

if K.image_data_format() == 'channels_first':
    input_shape = (3, img_width, img_height)
else:
    input_shape = (img_width, img_height, 3)

model = Sequential()
model.add(Conv2D(16, (3, 3), input_shape=input_shape))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(32, (3, 3)))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Flatten())
model.add(Dense(32))
model.add(Activation('relu'))
model.add(Dropout(0.3))
model.add(Dense(2))
model.add(Activation('softmax'))

model.summary()

```

```

Model: "sequential_2"

```

Layer (type)	Output Shape	Param #
conv2d_3 (Conv2D)	(None, 48, 48, 16)	448
activation_5 (Activation)	(None, 48, 48, 16)	0
max_pooling2d_3 (MaxPooling2D)	(None, 24, 24, 16)	0
conv2d_4 (Conv2D)	(None, 22, 22, 32)	4640
activation_6 (Activation)	(None, 22, 22, 32)	0
max_pooling2d_4 (MaxPooling2D)	(None, 11, 11, 32)	0
flatten_2 (Flatten)	(None, 3872)	0
dense_3 (Dense)	(None, 32)	123936
activation_7 (Activation)	(None, 32)	0
dropout_2 (Dropout)	(None, 32)	0
dense_4 (Dense)	(None, 2)	66
activation_8 (Activation)	(None, 2)	0

```

Total params: 129,090
Trainable params: 129,090
Non-trainable params: 0

```

```

model.compile(loss="sparse_categorical_crossentropy",
              optimizer="adam",
              metrics=["accuracy"])

train_datagen = ImageDataGenerator(
    rescale=1. / 255,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True)

test_datagen = ImageDataGenerator(rescale=1. / 255)

train_generator = train_datagen.flow_from_directory(
    train_data_dir,
    target_size=(img_width, img_height),
    batch_size=batch_size,
    class_mode='binary')

validation_generator = test_datagen.flow_from_directory(
    validation_data_dir,
    target_size=(img_width, img_height),
    batch_size=batch_size,
    class_mode='binary')

history = model.fit_generator(
    train_generator,
    steps_per_epoch=nb_train_samples // batch_size,
    epochs=epochs,
    validation_data=validation_generator,
    validation_steps=nb_validation_samples // batch_size)

model.save_weights('')

    Found 2894 images belonging to 2 classes.
    Found 290 images belonging to 2 classes.
    Epoch 1/8
    14/14 [=====] - 67s 5s/step - loss: 0.5974 - accuracy: 0.6841 - val_loss: 0.4457 - val_accuracy: 0.8100
    Epoch 2/8
    14/14 [=====] - 66s 5s/step - loss: 0.4121 - accuracy: 0.8215 - val_loss: 0.2450 - val_accuracy: 0.9222
    Epoch 3/8
    14/14 [=====] - 68s 5s/step - loss: 0.2782 - accuracy: 0.8921 - val_loss: 0.1835 - val_accuracy: 0.9450
    Epoch 4/8
    14/14 [=====] - 63s 5s/step - loss: 0.1848 - accuracy: 0.9382 - val_loss: 0.1348 - val_accuracy: 0.9444
    Epoch 5/8
    14/14 [=====] - 66s 5s/step - loss: 0.1305 - accuracy: 0.9584 - val_loss: 0.1084 - val_accuracy: 0.9650
    Epoch 6/8
    14/14 [=====] - 69s 5s/step - loss: 0.0971 - accuracy: 0.9684 - val_loss: 0.1640 - val_accuracy: 0.9444
    Epoch 7/8
    14/14 [=====] - 72s 5s/step - loss: 0.0869 - accuracy: 0.9739 - val_loss: 0.0545 - val_accuracy: 0.9850
    Epoch 8/8
    14/14 [=====] - 67s 5s/step - loss: 0.0668 - accuracy: 0.9814 - val_loss: 0.0615 - val_accuracy: 0.9778

model.save('')

model_json = model.to_json()
with open("model.json", "w") as json_file : json_file.write(model_json)

model.save_weights("")
print("Saved model to disk")

model.save('')

    Saved model to disk

print(history.history.keys())
plt.figure(1)
plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])
plt.title('model accuracy')
plt.ylabel('accuracy')
plt.xlabel('epoch')
plt.legend(['train', 'validation'], loc='upper left')

```

```
dict_keys(['val_loss', 'val_accuracy', 'loss', 'accuracy'])
<matplotlib.legend.Legend at 0x15097496d48>

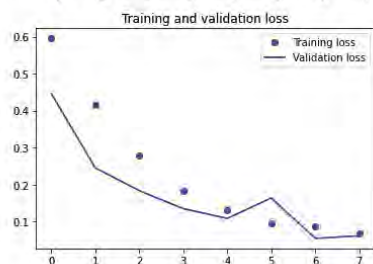
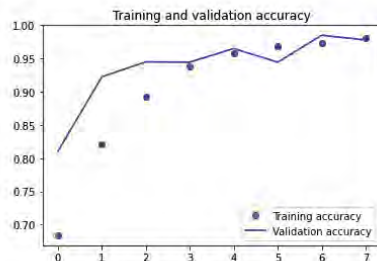
1.00
0.95
0.90
0.85
accuracy
train
validation

import matplotlib.pyplot as plt
%matplotlib inline

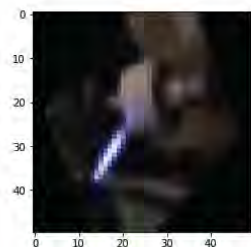
accuracy = history.history['accuracy']
val_accuracy = history.history['val_accuracy']
loss = history.history['loss']
val_loss = history.history['val_loss']
epochs = range(len(accuracy))

plt.plot(epochs, accuracy, 'bo', label='Training accuracy')
plt.plot(epochs, val_accuracy, 'b', label='Validation accuracy')
plt.title('Training and validation accuracy')
plt.legend()
plt.figure()

plt.plot(epochs, loss, 'bo', label='Training loss')
plt.plot(epochs, val_loss, 'b', label='Validation loss')
plt.title('Training and validation loss')
plt.legend()
plt.show()
```



```
test_img = plt.imread("")
from skimage.transform import resize
my_image_resized = resize(test_img, (img_width, img_height, 3))
img = plt.imshow(my_image_resized)
```



```

from keras import models
# extracts the outputs of the top 8 layers
layer_outputs = [layer.output for layer in model.layers[:8]]

# creates a model that will return these outputs, given the model input
activation_model = models.Model(input=model.input, output=layer_outputs)

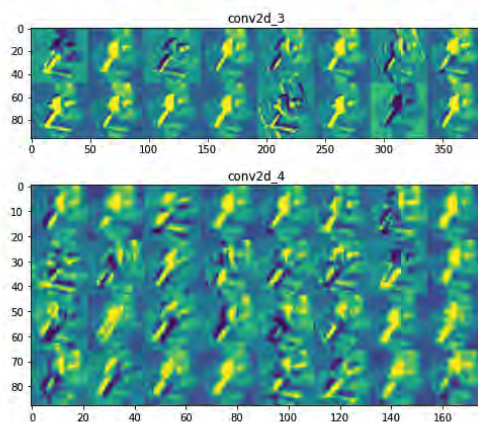
# returns a list of Numpy arrays: one array per layer activation
activations = activation_model.predict(np.array([my_image_resized,]))

# activation of the 1st convolution layer
first_layer_activation = activations[1]

# display the 4th channel of the activation of the 1st layer of the original model
plt.matshow(first_layer_activation[0, :, :, 4], cmap='viridis')

layer_names = []
for layer in model.layers[:-1]:
    layer_names.append(layer.name)
images_per_row = 8
for layer_name, layer_activation in zip(layer_names, activations):
    if layer_name.startswith('conv'):
        n_features = layer_activation.shape[-1]
        size = layer_activation.shape[1]
        n_cols = n_features // images_per_row
        display_grid = np.zeros((size * n_cols, images_per_row * size))
        for col in range(n_cols):
            for row in range(images_per_row):
                channel_image = layer_activation[0, :, :, col * images_per_row + row]
                channel_image -= channel_image.mean()
                channel_image /= channel_image.std()
                channel_image *= 64
                channel_image += 128
                channel_image = np.clip(channel_image, 0, 255).astype('uint8')
                display_grid[col * size : (col + 1) * size,
                             row * size : (row + 1) * size] = channel_image
        scale = 1. / size
        plt.figure(figsize=(scale * display_grid.shape[1],
                             scale * display_grid.shape[0]))
        plt.title(layer_name)
        plt.grid(False)
        plt.imshow(display_grid, aspect='auto', cmap='viridis')

```



```

my_image = plt.imread("")

from skimage.transform import resize

my_image_resized = resize(my_image, (img_width, img_height, 3))

img = plt.imshow(my_image_resized)

```





```

my_image_resized = resize(my_image, (img_width,img_height,3))

import numpy as np
probabilities = model.predict(np.array([my_image_resized,]))

number_to_class = ['no powder', 'powder']
index = np.argsort(probabilities[0,:])
print("Most likely class:", number_to_class[index[1]], "-- Probability:", probabilities[0,index[1]])
print("Second most likely class:", number_to_class[index[0]], "-- Probability:", probabilities[0,index[0]])

Most likely class: no powder -- Probability: 0.88276994
Second most likely class: powder -- Probability: 0.11723009

```

Appendix C: Image Similarity Implementation Code

Implementation Code Link:

<https://colab.research.google.com/drive/1Lgrpx3x5B6ds3cEPR4dmiTVmSbrad3hM?usp=sharing>

```
from skimage import measure
import matplotlib.pyplot as plt
import numpy as np
import cv2

def mse(imageA, imageB):

    err = np.sum((imageA.astype("float") - imageB.astype("float")) ** 2)
    err /= float(imageA.shape[0] * imageA.shape[1])
    return err

def compare_images(imageA, imageB, title):
    # setup the figure
    fig = plt.figure(title)
    plt.suptitle("MSE: %.2f, SSIM: %.2f" % (m, s))
    # show first image
    ax = fig.add_subplot(1, 2, 1)
    plt.imshow(imageA, cmap = plt.cm.gray)
    plt.axis("off")
    # show the second image
    ax = fig.add_subplot(1, 2, 2)
    plt.imshow(imageB, cmap = plt.cm.gray)
    plt.axis("off")
    # show the images
    plt.show()

HEA1 = cv2.imread("")
HEA2 = cv2.imread("")
HEA3 = cv2.imread("")
HEA4 = cv2.imread("")

HEA1 = cv2.cvtColor(HEA1, cv2.COLOR_BGR2GRAY)
HEA2 = cv2.cvtColor(HEA2, cv2.COLOR_BGR2GRAY)
HEA3 = cv2.cvtColor(HEA3, cv2.COLOR_BGR2GRAY)
HEA4 = cv2.cvtColor(HEA4, cv2.COLOR_BGR2GRAY)

fig = plt.figure("Images")
images = ("HEA_1", HEA1), ("HEA_2", HEA2), ("HEA_3", HEA3), ("HEA_4", HEA4)

for (i, (name, image)) in enumerate(images):

    ax = fig.add_subplot(1, 5, i + 1)
    ax.set_title(name)
    plt.imshow(image, cmap = plt.cm.gray)
    plt.axis("off")

plt.show()

compare_images(HEA2, HEA1, "Conforming vs. Optimised")
compare_images(HEA2, HEA3, "Conforming vs. Medium_Temp")
compare_images(HEA2, HEA4, "Conforming vs. High_Temp")
```

Appendix D: Coating Microstructure Defect Classification Code

Implementation Code Link:

<https://colab.research.google.com/drive/1QxlbFO7YxR4bQlhxsSmvbUye1nlZFPPS?usp=sharing>

```

from keras.preprocessing.image import ImageDataGenerator, load_img, img_to_array
from keras.models import Sequential
from keras.layers import Conv2D, MaxPooling2D
from keras.layers import Activation, Dropout, Flatten, Dense
from keras import backend as K
import matplotlib.pyplot as plt
from keras.models import model_from_json
from keras.models import load_model
import os
import cv2
import PIL
import numpy as np
import pandas as pd
import datetime
import tensorflow as tf

# dimensions of our images.
img_width, img_height = 100, 100
train_data_dir = 'data/train'
validation_data_dir = 'data/validation'
nb_train_samples = 95
nb_validation_samples = 24
epochs = 45
batch_size = 95

if K.image_data_format() == 'channels_first':
    input_shape = (3, img_width, img_height)
else:
    input_shape = (img_width, img_height, 3)

model = Sequential()
model.add(Conv2D(16, (3, 3), input_shape=input_shape))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(32, (3, 3)))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Flatten())
model.add(Dense(32))
model.add(Activation('relu'))
model.add(Dropout(0.12))
model.add(Dense(3))
model.add(Activation('softmax'))

model.compile(loss="sparse_categorical_crossentropy",
              optimizer="adam",
              metrics=["accuracy"])

train_datagen = ImageDataGenerator(
    rescale=1. / 255,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True)

test_datagen = ImageDataGenerator(rescale=1. / 255)

train_generator = train_datagen.flow_from_directory(
    train_data_dir,
    target_size=(img_width, img_height),
    batch_size=batch_size,
    class_mode='binary')

validation_generator = test_datagen.flow_from_directory(
    validation_data_dir,
    target_size=(img_width, img_height),
    batch_size=batch_size,
    class_mode='binary')

history = model.fit_generator(
    train_generator,
    steps_per_epoch=nb_train_samples // batch_size,
    epochs=epochs,
    validation_data=validation_generator,
    validation_steps=nb_validation_samples // batch_size)

```

```
model.save_weights('')
```

```
model.summary()
```

Model: "sequential_1"

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 98, 98, 16)	448
activation_1 (Activation)	(None, 98, 98, 16)	0
max_pooling2d_1 (MaxPooling2D)	(None, 49, 49, 16)	0
conv2d_2 (Conv2D)	(None, 47, 47, 32)	4640
activation_2 (Activation)	(None, 47, 47, 32)	0
max_pooling2d_2 (MaxPooling2D)	(None, 23, 23, 32)	0
flatten_1 (Flatten)	(None, 16928)	0
dense_1 (Dense)	(None, 32)	541728
activation_3 (Activation)	(None, 32)	0
dropout_1 (Dropout)	(None, 32)	0
dense_2 (Dense)	(None, 3)	99
activation_4 (Activation)	(None, 3)	0
Total params: 546,915		
Trainable params: 546,915		
Non-trainable params: 0		

```
model.save('')
```

```
model_json = model.to_json()
```

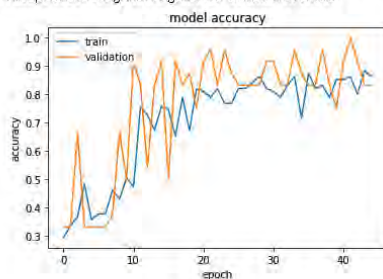
```
with open("model.json", "w") as json_file : json_file.write(model_json)
```

```
model.save_weights("")
```

```
print("Saved model to disk")
```

```
model.save('')
```

```
dict_keys(['val_loss', 'val_accuracy', 'loss', 'accuracy'])
<matplotlib.legend.Legend at 0x1a63fa72488>
```



```

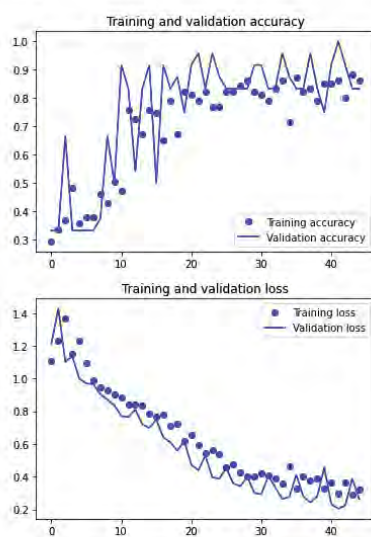
import matplotlib.pyplot as plt
%matplotlib inline

accuracy = history.history['accuracy']
val_accuracy = history.history['val_accuracy']
loss = history.history['loss']
val_loss = history.history['val_loss']
epochs = range(len(accuracy))

plt.plot(epochs, accuracy, 'bo', label='Training accuracy')
plt.plot(epochs, val_accuracy, 'b', label='Validation accuracy')
plt.title('Training and validation accuracy')
plt.legend()
plt.figure()

plt.plot(epochs, loss, 'bo', label='Training loss')
plt.plot(epochs, val_loss, 'b', label='Validation loss')
plt.title('Training and validation loss')
plt.legend()
plt.show()

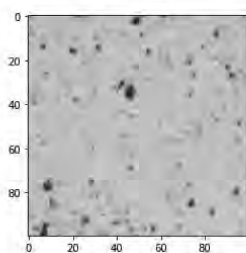
```



```

test_img = plt.imread("")
from skimage.transform import resize
my_image_resized = resize(test_img, (img_width, img_height, 3))
img = plt.imshow(my_image_resized)

```



```

from keras import models
layer_outputs = [layer.output for layer in model.layers[:8]]

activation_model = models.Model(input=model.input, output=layer_outputs)

activations = activation_model.predict(np.array([my_image_resized,]))

first_layer_activation = activations[0]

plt.matshow(first_layer_activation[0, :, :, 4], cmap='viridis')

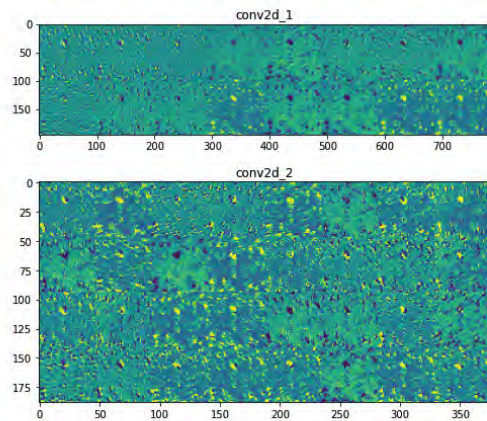
```

```

layer_names = []
for layer in model.layers[:-1]:
    layer_names.append(layer.name)
images_per_row = 8
for layer_name, layer_activation in zip(layer_names, activations):
    if layer_name.startswith('conv'):
        n_features = layer_activation.shape[-1]
        size = layer_activation.shape[1]
        n_cols = n_features // images_per_row
        display_grid = np.zeros((size * n_cols, images_per_row * size))
        for col in range(n_cols):
            for row in range(images_per_row):
                channel_image = layer_activation[0, :, :, col * images_per_row + row]
                channel_image -= channel_image.mean()
                channel_image /= channel_image.std()
                channel_image *= 64
                channel_image += 128
                channel_image = np.clip(channel_image, 0, 255).astype('uint8')
                display_grid[col * size : (col + 1) * size,
                                row * size : (row + 1) * size] = channel_image

        scale = 1. / size
        plt.figure(figsize=(scale * display_grid.shape[1],
                            scale * display_grid.shape[0]))
        plt.title(layer_name)
        plt.grid(False)
        plt.imshow(display_grid, aspect='auto', cmap='viridis')

```



```

my_image = plt.imread("")

from skimage.transform import resize

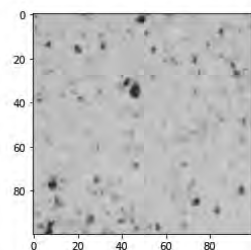
my_image_resized = resize(my_image, (img_width, img_height, 3))

img = plt.imshow(my_image_resized)

my_image_resized = resize(my_image, (img_width, img_height, 3))

import numpy as np
probabilities = model.predict(np.array([my_image_resized,]))

```



```
number_to_class = ['High Oxides', 'High Porosity', 'Conforming']
index = np.argsort(probabilities[0,:])
print("Most likely class:", number_to_class[index[2]], "-- Probability:", probabilities[0,index[2]])
print("Second most likely class:", number_to_class[index[1]], "-- Probability:", probabilities[0,index[1]])
print("Most likely class:", number_to_class[index[0]], "-- Probability:", probabilities[0,index[0]])

Most likely class: High Porosity -- Probability: 0.7938481
Second most likely class: High Oxides -- Probability: 0.15190212
Most likely class: Conforming -- Probability: 0.05424972
```

Appendix E: Super Resolution Training Model Implementation Code

Implementation Code Link:

https://colab.research.google.com/drive/19-Z-rhBuRt-n1kPwxH2bbohD_XSlagZ?usp=sharing

```

import tensorflow as tf
import os
import math
import numpy as np
from tensorflow import keras
from tensorflow.keras import layers
from tensorflow.keras.preprocessing.image import load_img
from tensorflow.keras.preprocessing.image import array_to_img
from tensorflow.keras.preprocessing.image import img_to_array
from tensorflow.keras.preprocessing import image_dataset_from_directory
from IPython.display import display

dataset_url = ""
data_dir = keras.utils.get_file(origin=dataset_url, fname="", untar=True)
root_dir = os.path.join(data_dir, "")

crop_size = 200
upscale_factor = 2
input_size = crop_size // upscale_factor
batch_size = 8

train_ds = image_dataset_from_directory(
    root_dir,
    batch_size=batch_size,
    image_size=(crop_size, crop_size),
    validation_split=0.2,
    subset="training",
    seed=1337,
    label_mode=None,
)

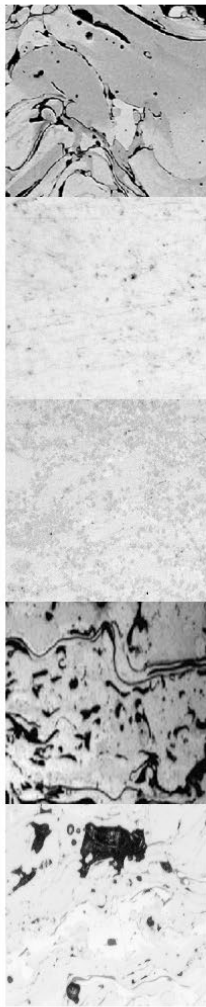
valid_ds = image_dataset_from_directory(
    root_dir,
    batch_size=batch_size,
    image_size=(crop_size, crop_size),
    validation_split=0.2,
    subset="validation",
    seed=1337,
    label_mode=None,
)

def scaling(input_image):
    input_image = input_image / 255.0 #was 255.0
    return input_image

# Scale from (0, 255) to (0, 1)
train_ds = train_ds.map(scaling)
valid_ds = valid_ds.map(scaling)

for batch in train_ds.take(1):
    for img in batch:
        display(array_to_img(img))

```



```
dataset = os.path.join(root_dir, "")
test_path = os.path.join(dataset, "")

test_img_paths = sorted(
    [
        os.path.join(test_path, fname)
        for fname in os.listdir(test_path)
        if fname.endswith(".jpg")
    ]
)
```

```

def process_input(input, input_size, upscale_factor):
    input = tf.image.rgb_to_yuv(input)
    last_dimension_axis = len(input.shape) - 1
    y, u, v = tf.split(input, 3, axis=last_dimension_axis)
    return tf.image.resize(y, [input_size, input_size], method="area")

def process_target(input):
    input = tf.image.rgb_to_yuv(input)
    last_dimension_axis = len(input.shape) - 1
    y, u, v = tf.split(input, 3, axis=last_dimension_axis)
    return y

train_ds = train_ds.map(
    lambda x: (process_input(x, input_size, upscale_factor), process_target(x))
)
train_ds = train_ds.prefetch(buffer_size=32)

valid_ds = valid_ds.map(
    lambda x: (process_input(x, input_size, upscale_factor), process_target(x))
)
valid_ds = valid_ds.prefetch(buffer_size=32)

for batch in train_ds.take(1):
    for img in batch[0]:
        display(array_to_img(img))
    for img in batch[1]:
        display(array_to_img(img))

```



```

def get_model(upscale_factor=3, channels=1):
    conv_args = {
        "activation": "relu",
        "kernel_initializer": "Orthogonal",
        "padding": "same",
    }
    inputs = keras.Input(shape=(None, None, channels))
    x = layers.Conv2D(64, 5, **conv_args)(inputs)
    x = layers.Conv2D(64, 3, **conv_args)(x)
    x = layers.Conv2D(32, 3, **conv_args)(x)
    x = layers.Conv2D(channels * (upscale_factor ** 2), 3, **conv_args)(x)
    outputs = tf.nn.depth_to_space(x, upscale_factor)

    return keras.Model(inputs, outputs)

import matplotlib.pyplot as plt
from mpl_toolkits.axes_grid1.inset_locator import zoomed_inset_axes
from mpl_toolkits.axes_grid1.inset_locator import mark_inset
import PIL

def plot_results(img, prefix, title):
    """Plot the result with zoom-in area."""
    img_array = img_to_array(img)
    img_array = img_array.astype("float32") / 255.0

    fig, ax = plt.subplots()
    im = ax.imshow(img_array[:-1], origin="lower")

    plt.title(title)

    axins = zoomed_inset_axes(ax, 2, loc=2)
    axins.imshow(img_array[:-1], origin="lower")

    x1, x2, y1, y2 = 200, 300, 40, 80

    axins.set_xlim(x1, x2)

    axins.set_ylim(y1, y2)

    mark_inset(ax, axins, loc1=1, loc2=3, fc="none", ec="blue")
    plt.savefig(str(prefix) + "-" + title + ".png")
    plt.show()

def get_lowres_image(img, upscale_factor):
    """Return low-resolution image to use as model input."""
    return img.resize(
        (img.size[0] // upscale_factor, img.size[1] // upscale_factor),
        PIL.Image.BICUBIC,
    )

def upscale_image(model, img):
    """Predict the result based on input image and restore the image as RGB."""
    ycbcr = img.convert("YCbCr")
    y, cb, cr = ycbcr.split()
    y = img_to_array(y)
    y = y.astype("float32") / 255.0

    input = np.expand_dims(y, axis=0)
    out = model.predict(input)

    out_img_y = out[0]
    out_img_y *= 255.0

    out_img_y = out_img_y.clip(0, 255)
    out_img_y = out_img_y.reshape((np.shape(out_img_y)[0], np.shape(out_img_y)[1]))
    out_img_y = PIL.Image.fromarray(np.uint8(out_img_y), mode="L")
    out_img_cb = cb.resize(out_img_y.size, PIL.Image.BICUBIC)
    out_img_cr = cr.resize(out_img_y.size, PIL.Image.BICUBIC)
    out_img = PIL.Image.merge("YCbCr", (out_img_y, out_img_cb, out_img_cr)).convert(
        "RGB"
    )
    return out_img

```

```

class ESPCNCallback(keras.callbacks.Callback):
    def __init__(self):
        super(ESPCNCallback, self).__init__()
        self.test_img = get_lowres_image(load_img(test_img_paths[0]), upscale_factor)

    def on_epoch_begin(self, epoch, logs=None):
        self.psnr = []

    def on_epoch_end(self, epoch, logs=None):
        print("Mean PSNR for epoch: %.2f" % (np.mean(self.psnr)))
        if epoch % 20 == 0:
            prediction = upscale_image(self.model, self.test_img)
            plot_results(prediction, "epoch-" + str(epoch), "prediction")

    def on_test_batch_end(self, batch, logs=None):
        self.psnr.append(10 * math.log10(1 / logs["loss"]))

early_stopping_callback = keras.callbacks.EarlyStopping(monitor="loss", patience=10)

checkpoint_filepath = ""

model_checkpoint_callback = keras.callbacks.ModelCheckpoint(
    filepath=checkpoint_filepath,
    save_weights_only=True,
    monitor="loss",
    mode="min",
    save_best_only=True,
)

model = get_model(upscale_factor=upscale_factor, channels=1)
model.summary()

callbacks = [ESPCNCallback(), early_stopping_callback, model_checkpoint_callback]
loss_fn = keras.losses.MeanSquaredError()
optimizer = keras.optimizers.Adam(learning_rate=0.001)

Model: "functional_47"
-----
Layer (type)                 Output Shape              Param #
-----
input_24 (InputLayer)       [(None, None, None, 1)]  0
-----
conv2d_92 (Conv2D)          (None, None, None, 64)   1664
-----
conv2d_93 (Conv2D)          (None, None, None, 64)   36928
-----
conv2d_94 (Conv2D)          (None, None, None, 32)   18464
-----
conv2d_95 (Conv2D)          (None, None, None, 4)    1156
-----
tf_op_layer_DepthToSpace_23 [(None, None, None, 1)]  0
-----
Total params: 58,212
Trainable params: 58,212
Non-trainable params: 0
-----

epochs = 100

model.compile(
    optimizer=optimizer, loss=loss_fn,
)

model.fit(
    train_ds, epochs=epochs, callbacks=callbacks, validation_data=valid_ds, verbose=2
)

model.load_weights(checkpoint_filepath)

```

Appendix: Python Script for spectrogram overlay in video frames

```
1  import cv2
2  import numpy as np
3  import pyaudio
4  import matplotlib.pyplot as plt
5  from scipy import signal
6
7  # Open a connection to the webcam
8  cap = cv2.VideoCapture(0)
9
10 # Set up the audio stream
11 p = pyaudio.PyAudio()
12 stream = p.open(format=pyaudio.paInt16, channels=1, rate=44100, input=True,
13                 frames_per_buffer=1024)
14
15 while(True):
16     # Capture frame-by-frame
17     ret, frame = cap.read()
18
19     # Capture audio
20     audio_data = np.frombuffer(stream.read(1024), dtype=np.int16)
21
22     # Compute the spectrogram
23     f, t, Sxx = signal.spectrogram(audio_data, 44100)
24
25     # Plot the spectrogram
26     plt.pcolormesh(t, f, 10 * np.log10(Sxx))
27     plt.ylabel('Frequency [Hz]')
28     plt.xlabel('Time [sec]')
29     plt.colorbar()
30
31     # Convert the plot to an image
32     plt.draw()
33     canvas = plt.gcf().canvas
34     img = np.frombuffer(canvas.tostring_rgb(), dtype=np.uint8)
35     img = img.reshape(canvas.get_width_height()[::-1] + (3,))
36     img = cv2.cvtColor(img, cv2.COLOR_RGB2BGR)
37
38     # Resize the image to fit the corner of the video frame
39     img = cv2.resize(img, (frame.shape[1]//4, frame.shape[0]//4))
40
41     # Overlay the images
42     frame[0:img.shape[0], frame.shape[1]-img.shape[1]:frame.shape[1]] = img
43
44     # Display the resulting frame
45     cv2.imshow('Frame', frame)
46
47     plt.close()
48
49     # Break the loop
50     if cv2.waitKey(1) & 0xFF == ord('q'):
51         break
52
53 # When everything done, release the capture
54 cap.release()
55 cv2.destroyAllWindows()
56 stream.stop_stream()
57 stream.close()
58 p.terminate()
```