


## Article

# Software (GUI/APP) for Developing AI-Based Models Capable of Predicting Load-Displacement Curve and AFM Image during Nanoindentation

Rajesh Jha \*  and Arvind Agarwal 

Department of Mechanical and Materials Engineering, College of Engineering and Computing, Florida International University, 10555 West Flagler Street, Miami, FL 33174, USA; agarwala@fiu.edu

\* Correspondence: rjha@fiu.edu

**Abstract:** During nanoindentation tests, the load-displacement curve is used for estimating mechanical properties, while an indent image obtained through atomic force microscopy (AFM) is used for studying deformation of a material. We present a computational platform for developing artificial intelligence-based models for predicting indentation depth (load-displacement curve) and AFM image as a function of test parameters like maximum applied load, loading rate, and holding time. A user can directly use machine generated data in text (.txt) and hierarchical data format (HDF, hdf) format for developing the AI-based models for indentation depth and AFM image, respectively. The software was tested on three different coatings/materials for indentation depth: heat-treated (HT) sample of cold sprayed aluminum-based bulk metallic glass (Al-BMG) coating, carbon nanotube reinforced aluminum composite (Al-5CNT) coating, and spark-plasma-sintered hydroxyapatite (SPS HA) sample. For AFM imaging, a heat-treated (HT) sample of cold sprayed aluminum-based bulk metallic glass (Al-BMG) coating was considered. Correlation or *R*-values are close to 1 for all the models developed in this work. Predicted load-displacement curve and AFM image are in good agreement with the experimental findings. Our approach will be helpful in virtual simulation of load-displacement curves and AFM indent images for a large number of new test parameters, thus significantly reducing the number of indents needed for characterizing/analyzing a material.

**Keywords:** coating; nanoindentation; atomic force microscopy (AFM); hierarchical data format (HDF); artificial intelligence (AI); MATLAB APP; graphical user interface (GUI)



**Citation:** Jha, R.; Agarwal, A. Software (GUI/APP) for Developing AI-Based Models Capable of Predicting Load-Displacement Curve and AFM Image during Nanoindentation. *Coatings* **2021**, *11*, 299. <https://doi.org/10.3390/coatings11030299>

Academic Editor: Stefano Farris

Received: 14 February 2021

Accepted: 28 February 2021

Published: 5 March 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

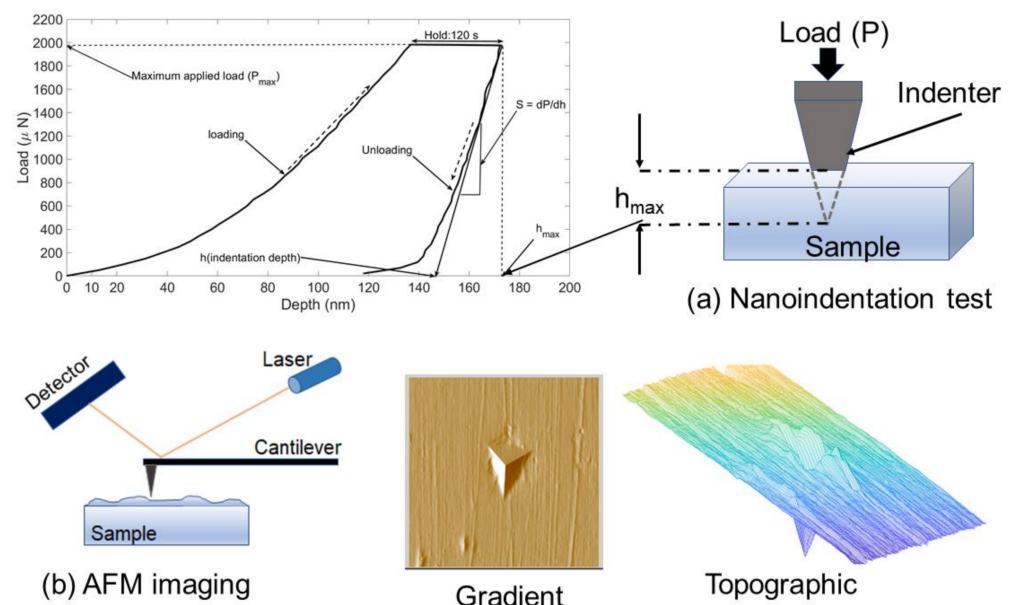


**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Nanoindentation is widely used for determining mechanical properties of materials by analyzing the load-displacement curve [1,2], while atomic force microscopy (AFM) is an essential tool for analyzing the deformation of material during indentation [3,4]. In order to perform experiments on a new material, an experimentalist relies on their experience or refers to the literature on tests performed on similar systems, and accordingly designs their approach regarding applied load and holding time for similar systems [5–7]. This approach utilizes the expertise of an experienced experimentalist to come up with a set of test conditions. In cases where the expert thinks that more tests are needed, new nanoindentation test parameters are defined for another battery of iterative tests [6,7]. From the experiments and testing, a user needs load-displacement curves to study the material response and to estimate mechanical properties while they also check the indent image to study the deformation behavior [1,2,4]. An indenter tip can hit an uneven area or edges, the surface of the coating can fracture, or a load-displacement curve is not as per the expectation of the user [4,8]. In such cases, imaging helps the experimentalist in choosing the correct set of locations for experiments [4,8]. At present, this is the conventional approach and has been time tested and widely used by research groups around the world.

Figure 1 provides a schematic view of nanoindentation experiment and AFM imaging. A nanoindentation test and load-displacement curve generation is shown in Figure 1a. AFM imaging and generation of topographic and gradient image is shown in Figure 1b. There exists an interaction force between the AFM tip and the sample surface when the tip is close to the surface. During AFM imaging, this interaction force can be estimated. The AFM tip is attached at the end of a cantilever and this cantilever is moved parallel to the surface in order to estimate the surface profile. Though the cantilever moves along the x-y direction, the system assigns “FORWARD” and “REVERSE” terminology based on the starting point from where the cantilever starts moving. A laser system tracks the movement of the cantilever and this laser detection system is capable of estimating the interaction force between the AFM tip and the surface. During AFM imaging, surface roughness affects the interaction force between the AFM tip and the surface. For the AFM system, a sensor setpoint is defined in the beginning. A laser detection system compares the measured signal with the sensor setpoint and when it detects a change, it resets the distance between the AFM tip and the surface in z-direction. The difference between the measured signal and the sensor setpoint is known as “error signal”. This error signal is assigned a zero value by changing the distance between the AFM tip and the surface. Thus, AFM imaging records two types of signal, (1) z-correction and (2) error signal. AFM imaging provides with two types of surface profile or image, namely “TOPOGRAPHY” and “GRADIENT”, where topography is the z-correction, while gradient is the error signal [8].



**Figure 1.** Schematic representation of (a) nanoindentation test, (b) atomic force microscopy (AFM) imaging.

### 1.1. Modeling and Simulations: Current State of the Art

Nanoindentation tests have been simulated in the past under the framework of finite element approach (FEA) and molecular dynamics (MD) approach, but both the approaches are computationally expensive, and a user may need supercomputer access [9–14]. Since 2011, the USA government has invested significant resources in the integrated computational materials engineering (ICME) approach as part of the Materials Genome Initiative (MGI). The MGI/ICME approach motivates professionals from academia, research labs, and industries to collaborate and focus on implementing computational tools in their work, in order to reduce the time between discovery and deployment of new materials [15]. Data generated from nanoindentation tests and FEA simulations have been analyzed through machine learning algorithms [16–19].



## 1.2. Challenges: Experiments, Modeling, and Simulations

One of the challenges with the conventional approach is that experiments are time-consuming and expensive, thus random experiments can be misleading and may cost a fortune, while simulation software may need either expensive license or supercomputer access or both. These limitations provide a congenial as well as challenging opportunity for data scientists to utilize their expertise and develop predictive models that can capture the complex correlations between various parameters an experimentalist varies during a nanoindentation test and results from a nanoindentation test.

Researchers at Ansys [20] and Bruker [21,22] have demonstrated the scope of utilizing data science algorithms for analyzing nanoindentation data, including data generated through atomic force microscopy. Researchers are proposing approaches which will be helpful in decreasing number of experiments required for analyzing a material or a coating [22,23]. Nanoindentation experiments and AFM mapping can be expensive and time-consuming [22]. Thus, any improvement in minimizing the number of experiments required for analyzing a material/coating will be helpful in serving the goal of the ICME/MGI initiative, which is decreasing the time period between the discovery of a new material and its deployment.

### 1.2.1. Proposed Computational Platform/Software

This motivated us to make an attempt to develop a computational framework that will include modules for developing models for predicting nanoindentation test results for new test condition. AI-based models will be capable of predicting indentation depth/load-displacement curve and AFM indent image. These models can be used as a screening tool prior to performing experiments. Thus, our approach has the potential to minimize the number of experiments required in analyzing/characterizing a coating/material. Our past experimental research on nanoindentation [24–28] has been coupled with our expertise in application of several concepts of artificial intelligence (AI) on data obtained from experiments and calculation of phase diagram (CALPHAD) approach [29,30] in framing our problem while developing this software (GUI/APP). GUI stands for graphical user interface. In MATLAB 2019 [31], the GUI is also referred to as APP.

### 1.2.2. Problem Formulation for Software Development

We have formulated our problem as follows:

- Software (GUI/APP) has independent modules for developing predictive models for indentation depth/load displacement curve and AFM indent image.
- Software (GUI/APP) can predict the outcomes of a nanoindentation test for new test conditions in form of indentation depth (load-displacement curve) and indent image (AFM).
- Software (GUI/APP) can be used on a regular desktop and laptop, thus avoiding significant investments in purchasing new computers.
- Computation time must be reasonable. We defined an upper limit of one hour for model development and prediction of nanoindentation test results for new test conditions.
- Develop a version of software that can be used for free.
- Software (GUI/APP) must have options for users who are experimentalists with limited exposure to AI algorithms, and also for users who are expert in AI.
- Provision for plotting load-displacement curve and AFM image from machine generated files in text and HDF format [32] respectively without pre-processing or even opening these files.

### 1.2.3. Software Testing: Case Studies

Thus, we present a software (GUI/APP) that will enable experimentalists to utilize the data from actual experiments and use our software for predicting indentation “depth” simultaneously plotting a load-displacement curve along with the AFM indent image

through our platform for new test conditions. Our approach utilizes raw data files generated by the machine in text (.txt) and hierarchical data format (HDF)5 (.hdf) files [33]. All of our software/model predictions are experimentally verified. For “depth” prediction, we tested our approach on: (1) cold sprayed aluminum-based bulk metallic glass (Al-BMG) coating: heat treated (HT) samples [26,27], (2) carbon nanotube reinforced aluminum composite (Al-5CNT) coating [24,25], and (3) spark-plasma-sintered hydroxyapatite (SPS HA) samples [28]. For AFM image, we chose (1) cold sprayed aluminum-based bulk metallic glass (Al-BMG) coating: heat treated (HT) samples [26,27]. After carefully analyzing the experimental and predicted results and from their own expertise in performing nano-indentation experiments, an experimentalist can significantly reduce the number of experiments performed by them by using our platform. This software (GUI/App) was developed in MATLAB programming language [31,33]. It can be used as a MATLAB APP for MATLAB users on any operating system. We have also prepared a standalone (free) software for Windows operating system for users without a MATLAB license.

### 1.3. Main Objective and Outcomes

The main objective of this work is to minimize the number of nanoindentation experiments to be performed while characterizing a coating/material. Through this software, machine generated raw data from a limited number of experiments can be utilized for developing AI-based models for indentation depth and AFM image. These models can then be used as a predictive tool for simulating experimental results generated by the machine for new test conditions. Correlation or regression ( $R$ -value) for the developed models are close to 1, and AI-based model predictions are in good agreement with the experimental findings. Through this work, we have demonstrated that even 9 nanoindentation experiments are sufficient for developing accurate AI-based models for indentation depth, while 18 sets of AFM image data provided with models can accurately predict the AFM image for new test conditions.

### Novelty

This is the first work of its kind in which a user can predict AFM image. AFM image has been used by researchers and correlated with experiments [20–22]. However, in this work, a user can predict an AFM image as a function of new test conditions/parameters, which is a novelty.

Regarding indentation depth, we have demonstrated that the randomness of experimental test conditions is helpful in developing accurate AI-based models from data from a limited number of experiments.

This software is not dependent on the type of material. It utilizes actual experimental data for developing AI-based models. Data can be of any material. Model development and testing of new experimental parameters can be done within an hour. Thus, a user can perform a predefined set of experiments and then analyze this data through our platform, simulate a few virtual experiments, and then decide if they want to perform new experiment. All this can be done in an hour.

## 2. Materials and Methods

In this work, we utilized data obtained from experiments performed by our group, which are also part of previously published papers [24–28]. This software uses raw data from the nano-indenter machine.

### 2.1. Machine Generated Data

Nanoindentation machine generates data corresponding to indentation depth and stores it in text (.txt) file format, while imaging data is stored in HDF5 (.hdf) file format.

### 2.1.1. Indentation Depth Data in Text (.txt) Format

Raw data in text (.txt) files contain information on “indentation depth”, “load”, and “time” data for each indent. For each indent, there can be a set of parameters like maximum load ( $\mu\text{N}$ ), loading rate ( $\mu\text{N/s}$ ), or time taken to reach maximum load(s) for experiments where a user sticks with a constant loading rate for several indents at a time.

### 2.1.2. AFM Image Data in HDF (.hdf) Format

Raw data for AFM indent image is stored in HDF5 (.hdf) file format. Imaging can be performed in either FORWARD or REVERSE direction or both [34]. For a case when imaging was performed in both FORWARD and REVERSE, system generates a (.BMP) file and stores data in four HDF [32,34] files with notation GF, GR, TF, and TR. GF stands for gradient forward, TF stands for topography forward, GR stands for gradient reverse, and TR stands for topography reverse [34]. Topography and gradient modes have been explained in detail in Section 1: Introduction [8].

HDF stands for hierarchical data format and is used for storing extremely large amounts of data in an organized way so that it can be interpreted by a wide range of commercial as well as non-commercial software platforms like MATLAB, Java, Scilab, Octave, R, Python, and Mathematica [32]. HDF suite is a product that came out of work initiated in 1987 by Graphics Foundations Task Force (GFTF) at the National Center for Supercomputing Applications (NCSA), at the University of Illinois at Urbana-Champaign, Urbana, Illinois. Currently, it is supported by The HDF Group, which is a non-profit corporation [32].

## 2.2. Case Studies

Case studies were prepared to test the prediction capability of models developed for indentation depth and AFM indent image. We have provided a brief description on materials and experimental test conditions. Cited publications will provide detailed information on the nanoindentation experiments [24–28]. In the main article, case study 2.2.1. is discussed in detail, while the case studies 2.2.2. and 2.2.3. have been included in Appendix A.

AI-based models for indentation depth were developed for all three case studies, while an AI-based model for AFM image was only developed for case study 2.2.1. For other case studies, available AFM image files were inadequate for developing AI-based models.

### 2.2.1. Cold Sprayed Aluminum-Based Bulk Metallic Glass (AL-BMG) Coating

In this work, the sample was heat-treated prior to the nanoindentation test. During nanoindentation tests, maximum applied load was set at 1000, 2000, and 4000  $\mu\text{N}$ . Time to reach maximum load was fixed at 5 s for all the cases, that is 5 s for loading and 5 s for unloading. Holding time was set at 0, 5, 10, 15, 30, 60, 120, and 240 s. Data obtained after performing the nanoindentation test can be described as follows:

- Indentation depth prediction and plotting load-displacement curve: a total of 24 indentation experiments were performed and 24 text files were generated.
- Indentation image prediction using image files in HDF (.hdf) format: a total of 24 indentation experiments were performed. Out of which, AFM indentation imaging was performed in 18 of these indentation tests. Imaging was performed in both FORWARD and REVERSE direction. Thus, there were 18 HDF files for each of GF, GR, TF, and TR corresponding to each of these indents.

### 2.2.2. Al-5CNT Coating

A total of nine tests were performed. Maximum applied load/peak load varied from 984–5936  $\mu\text{N}$ . Time to reach maximum load was 10 s, that is 10 s for loading and 10 s for unloading. In this problem, “loading rate” varied along with peak load from about 98–594  $\mu\text{N/s}$ . Holding time for the indenter was fixed at 5 s for all the tests.

- Indentation depth prediction and plotting load-displacement curve: a total of nine indentation experiments were performed and nine text files were generated.

### 2.2.3. Spark-Plasma-Sintered Hydroxyapatite (HA) Sample

The test was performed on two different matrices where peak load was set at 2500  $\mu\text{N}$  and holding time for the indenter was 3 s. Ten experiments were performed for each matrix.

- Indentation depth prediction and plotting load-displacement curve: a total of 10 indentation experiments were performed and 10 text files were generated for each matrix. Separate load-displacement curves were plotted for each matrix.

## 2.3. Software Development

This software was coded/developed in MATLAB 2019a [31,33]. Software development work was initially performed on a computer with Windows 10 Home operating system. System details are as follows:

- Processor (CPU): Intel® Core™ i7-4980HQ
- CPU Speed: 2.8 to 4 GHz w/Turbo Boost
- Cores: 4
- Memory: 32 GB DDR3L 1600 MHz.

### 2.3.1. Execution Time

In the current version, execution time is as follows for MATLAB 2019a [31,33] on a laptop computer with system specifications listed above:

#### Indentation Depth and Plotting Load-Displacement Curve

- Development of one artificial neural network (ANN) model: varies between 1 to 10 min (maximum).
- Prediction of depth: maximum 1 min.

#### AFM Indent Imaging

Imaging can be performed in FORWARD and REVERSE direction. For this software, it is advised to perform the imaging in either FORWARD or REVERSE direction or BOTH.

- Development of model for AFM indent image:
  - Maximum 30–35 min if all 4 HDF files are used, GF, GR, TF, and TR. That is when imaging was performed in both FORWARD and REVERSE direction.
  - Maximum 15–20 min (maximum) if 2 HDF files are used (GF, TF) or (GR, TR). This is for the case when imaging was performed in either only FORWARD direction or only REVERSE direction.
- Prediction of an AFM indent image:
  - Maximum 3–4 min for all 4 HDF files, GF, GR, TF, and TR.
  - Maximum 2–3 min if 2 HDF files are used (GF, TF) or (GR, TR).

### 2.3.2. File Name

For this software, the file name includes the value of experimental parameters like maximum applied load, holding time, loading rate, etc.

- TEXT files generated from the machine: the name of the file must include the load and holding time, like ABCD-2000-30-XX.TXT, for maximum applied load of 2000  $\mu\text{N}$ , and holding time of 30 s.
- HDF files generated from the machine: name of the file must include the load and holding time, like ABCD-2000-30-XX-GF.HDF, ABCD-2000-30-XX-GR.HDF, ABCD-2000-30-XX-TF.HDF, and ABCD-2000-30-XX-TR.HDF for maximum applied load of 2000  $\mu\text{N}$ , and holding time of 30 s.
- The numerical value of any other parameters can be similarly added to the file name.



- This format of file name will be helpful, and the software will directly extract the parameter values or test condition values from the name of these files. This way a user will not have to put extra effort in preparing a separate sheet with test conditions for the experiments performed.

### 2.3.3. Software Layout

Figure 2 shows the layout of our software (GUI/APP). Detailed instructions on the software (GUI/APP) are provided in the Supplementary file and the video abstract along with this article.

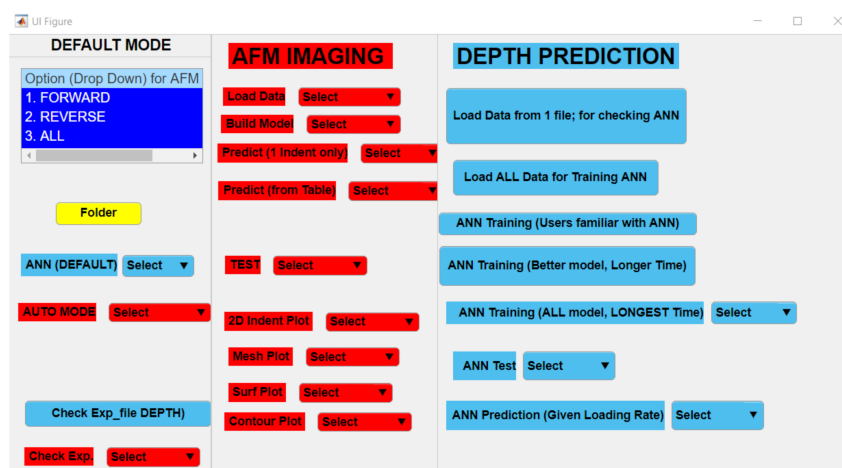


Figure 2. Layout of the software/app.

## 3. Results

This section has been divided into two subsections:

- Indentation depth (load-displacement curve) prediction
- Prediction of indent image (AFM).

### 3.1. Indentation Depth (Load-Displacement Curve) Prediction

In this section, we have explained model development for indentation depth as a function of experimental test conditions. Prediction results are presented in the form of load-displacement curve for the case study mentioned in the “Materials and Methods” section.

Case study 2.2.1.: heat-treated sample of cold sprayed aluminum-based bulk metallic glass (Al-BMG (HT)) coating [26,27]: nanoindentation machine generates results corresponding to indentation depth and the data is stored in a text file (.txt format). Each text file contains data for “depth”, “load”, and “time” in adjacent columns for a combination of test parameters. In this case study, test parameters were “maximum applied load” and “holding time” for three different loading rates. For each combination of “maximum applied load” and “holding time”, a text file is generated with about a thousand values generated by the machine for “depth”, “load”, and “time”. We identified four input parameters to affect indentation depth (“depth” from text file) of material in this case study. These parameters are user defined “maximum applied load” and “holding time” and two parameters (“load” and “time”) from the text file generated by the machine. Loading rate was not considered as an input parameter in this case study but has been included in the other case study.

Thus, we developed an ANN model for “depth” as a function of “maximum applied load”, “holding time”, “load”, and “time”.

#### 3.1.1. ANN Model Development in This Work

Data processing: as mentioned, each nanoindentation experiment generates results in a text (.txt) file. Each text (.txt) file contains about 1034 rows and data is recorded for “load”, “time”, and “depth”. In each of these text files, there can be additional text, blank rows, and

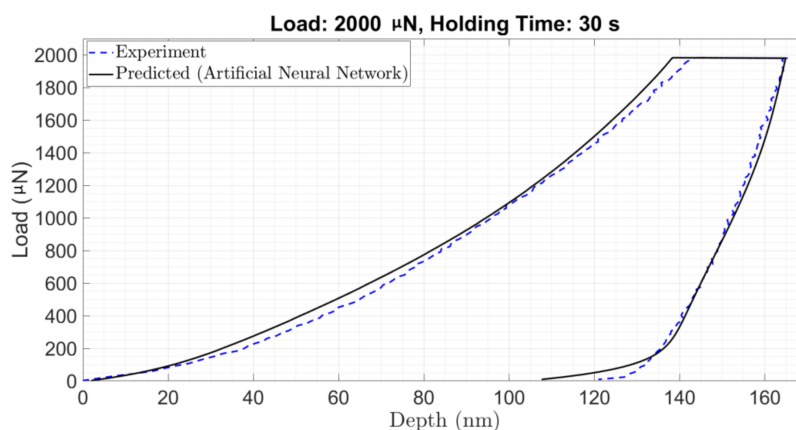
some rows where the sensor picked up negative “load” and “depth” data. These negative data along with blank rows and text need to be removed. Our approach automatically removes this during importing of data from the text (.txt) file.

We had 24 text files corresponding to 24 indents. Twenty-three files were used during model development for indentation “depth”. One file was kept for comparison of the load-displacement curve generated by machine and the one predicted by the ANN model for indentation “depth” (Figure 1). After preprocessing, each text file contains about 1000 rows. Thus, for 23 files we ended up with a large dataset of 23,420 rows, which is well suited for application of an artificial neural network (ANN). We used Deep Learning Toolbox™ (formerly Neural Network Toolbox™) in MATLAB [31,33] to develop the ANN model for predicting indentation “depth”. We used 70% of the data for training (16,394 samples), 15% for validation (3513 samples), and 15% for testing (3513 samples). In our current work, we started with 10 hidden neurons in the hidden layer and got optimal results for 15 hidden neurons. We chose Levenberg–Marquardt as the training algorithm [31,33]. The network was trained multiple times, so that we can observe similar or lower error in the testing set. Error values for ANN model development have been tabulated in Table 1.

**Table 1.** Error metrics for artificial neural network (ANN) model for indentation depth.

Data Set	Samples	MSE	R
Training	16,394	4.52912	0.999321
Validation	3513	4.58823	0.999311
Testing	3513	4.48478	0.999306

In Table 1, MSE stands for mean square error, which is the average of the squared differences between the outputs and targets. Lower MSE value means lower error. *R* stands for regression or correlation between the output from the model and the targets. An *R* value close to 1 means close relationship while an *R* value of 0 means random relationship. Figure 3 shows the comparison between the load-displacement curve obtained through the ANN model and the one plotted from data obtained from the experiments for heat-treated sample of cold sprayed aluminum-based bulk metallic glass (Al-BMG (HT)) coating. As mentioned before, we separated one file at the beginning, thus the ANN model was not trained on the data available in this text file. In Figure 2, we can observe that the ANN model is capable of catching the trends shown in the load-displacement curve for new data to which it was not exposed to while training. Error values are calculated through MATLAB software and stored along with the developed artificial neural network (ANN) model [31,33]. A user does not need to calculate these values separately.



**Figure 3.** Comparison: experiment vs. predicted for load of 2000  $\mu\text{N}$  and holding time of 30 s for heat-treated aluminum-based bulk metallic glass (Al-BMG (HT)) coating.

### 3.1.2. Discussions on ANN Model

We have tested/validated our approach of model development in two ways:

- Error metrics reported in Table 1: here, 15% of the data were assigned to the testing set. Data included in the testing set were not introduced to the ANN model during its development, but the ANN model performed well on the testing set as error metrics are similar for training, validation, and testing set reported in Table 1.
- Comparison between load-displacement curves obtained from the experiment and the curve predicted by ANN model: in this case too, these data were not introduced to the ANN model during training, but the ANN model performs well as it was able to capture the trends shown by the experiments (Figure 3). (We had 24 text files out of which 1 text file was separated in the beginning for this comparison. We mentioned that 23 files were used for model development).

An experimentalist utilizes load-displacement curves for estimating mechanical properties and can visually check if they are satisfied with the test from the nature of the load-displacement curve. Thus, error metrics and load-displacement curve will be helpful for selecting a model for predicting indentation depth (load-displacement curve) in the future.

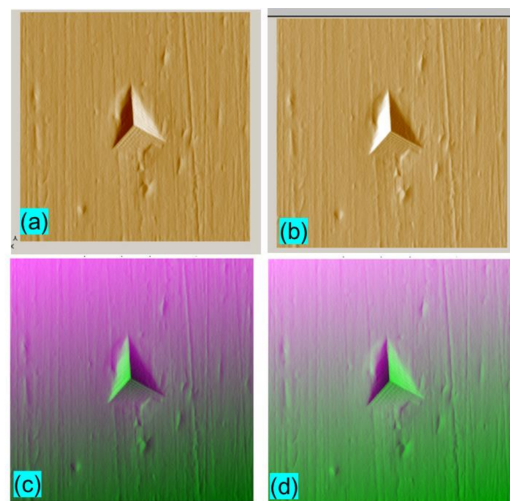
### 3.2. Prediction of Indent Image (AFM)

We used AFM data from the case study mentioned in “Materials and Methods” section.

Case Study (2.2.1.): prediction of indent image for heat-treated sample of cold sprayed aluminum-based bulk metallic glass (Al-BMG (HT)) coating [26,27]:

The machine stores data corresponding to the system-generated image in HDF file format [32]. For GE, GR, TF, and TR, data stored in an HDF file are in form of  $256 \times 256$  matrix. Thus, for each image file, we have 65,536 data points. We used data from these raw HDF files to develop AI-based models corresponding to the AFM image.

Figure 4a,b shows the AFM image obtained from the nanoindenter for heat-treated amorphous aluminum coating (BMG). Our platform can also be used for recreating the AFM image through HDF files generated by the machine. Figure 4c,d shows the image recreated by our software from the data extracted from the raw HDF files. We can observe that the system-generated images and the recreated image look similar. This motivated us to make an attempt to develop predictive models that will be helpful in predicting indent image (AFM) for new testing conditions.



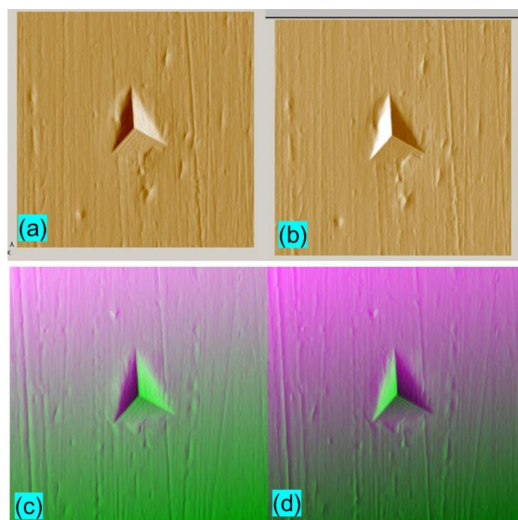
**Figure 4.** (a,b): System-generated image [26,27]; (c,d): recreated image from raw hierarchical data format (HDF) files generated by the system for maximum applicable load of 2000  $\mu\text{N}$  and 30 s holding time.

### 3.2.1. Model Development for AFM Image

As mentioned, 24 nanoindentation experiments were performed, out of which imaging was performed in 18 of these tests. We had 18 sets of files (4 HDF files in each set) corresponding to each of these tests. A total of 17 sets of files (4 HDF files each) were used during model development, and 1 set of files (4 HDF) was kept for comparison. We identified two input parameters that affect material deformation studied through indent image (AFM) for this case study. These parameters are user defined “maximum applied load” and “holding time”.

Thus, we developed a model for “AFM indent image” as a function of “maximum applied load” and “holding time” using data stored in the AFM imaging files in HDF (.hdf) format.

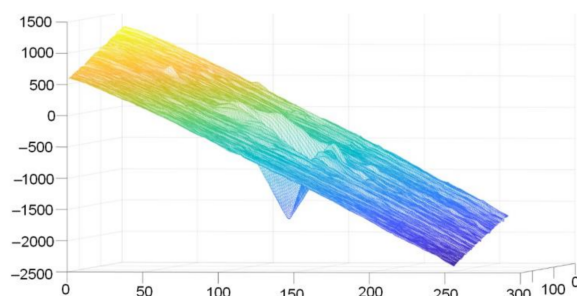
Model development takes about 15–40 min, while prediction of image takes about 2–4 min. Figure 5 shows the comparison between the images obtained from the machine (Figure 5a,b), and that predicted by our approach (Figure 5c,d) for a case for which data were not introduced to the model.



**Figure 5.** (a,b): System-generated image [26,27]; (c,d): predicted image (current work) for maximum applicable load of 2000  $\mu\text{N}$  and 30 s holding time.

From Figure 5, we can observe that the predicted indent looks similar to the indent generated from the machine. From the indentation depth model, we already have the predicted load-displacement curve. Thus, the image of our predicted indent can prove to be useful as it captures the morphology of the indent observed from the experiments.

Figure 6 shows the 3D topographical image of the indent predicted through our software for maximum applied load of 2000  $\mu\text{N}$  and 30 s holding time.

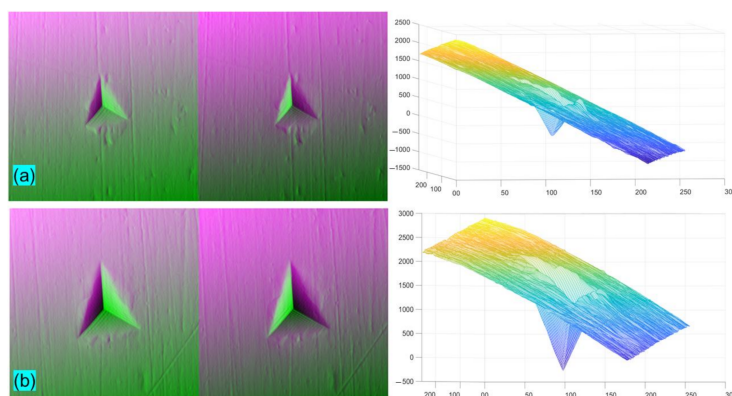


**Figure 6.** Predicted topographical image of indent for load 2000  $\mu\text{N}$ , holding time 30 s.



### 3.2.2. Testing an AFM Image Model for New Test Condition

We defined new test conditions: maximum applied load and holding time, for testing our model for the indent image (AFM). Figure 7 shows the predicted images (AFM) for new test conditions. We can observe that visually the indents are smaller for image in Figure 7a when compared to Figure 7b. In Figure 7a, load was 1730  $\mu\text{N}$  and time was 140 s, while in Figure 7b, load was 3900  $\mu\text{N}$ , and time was 230 s. From our experience, we can confirm that larger load and long holding time will result in larger indents.



**Figure 7.** Predicted 2D (forward and reverse) and 3D topographic image for new test parameters from the table provided by the user (a) Sl. No. 7: load: 1730  $\mu\text{N}$ , time: 140, and (b) Sl. No. 17: load: 3900  $\mu\text{N}$ , time: 230 s.

As we mentioned before, predicting indent image takes time, about 2–4 min for each new set of test parameters. Thus, we prepared an “Excel sheet” with a set of 20 new test parameters: maximum applied load and Holding time. Once models are developed, this software can be used to predict AFM indent images. A user needs to place this Excel sheet in the same folder and the software will predict the AFM indent images. Thus, “Sl.No.” in Figure 7a,b corresponds to the combination of new test parameters in the Excel sheet provided by the user. This way, a user can develop models for AFM indent image in 15–35 min, and then leave the software running in case they want to analyze a large number of AFM indent images for new test conditions.

In Figure 7, we can observe that the predicted AFM image looks realistic for new test parameters defined by the user. Our approach provides flexibility to the user in analyzing the AFM image in the form of 2D indent image and 3D indent profile of an indent for a large number of new test conditions (or parameters) based on a few experimental AFM images.

## 4. Discussion

This is a novel work for developing AI-based predictive models which can be used as a predictive tool for simulating experimental outcomes for new test parameters. AI-based model predictions are presented in the form of load-displacement curve and AFM indent image.

### 4.1. AI-Based Models for Indentation Depth

The main objective of this work is to minimize the number of experiments required for characterizing a coating/material through nanoindentation. Thus, this work is different when compared with AI-based articles on nanoindentation [16–19] as these articles are focused on determining nanomechanical properties from nanoindentation data.

In this work (Case Study (2.2.2.) in Appendix A), we have demonstrated that even a set of nine experiments is sufficient for developing AI-based models for “indentation depth” and subsequently generating load-displacement curve.

Experimental design and its effect on required number of experiments can be summarized as follows:

- Case study 2.2.1. cold sprayed aluminum-based bulk metallic glass (Al-BMG) coating [26,27]:
  - MSE ~4.5 and correlation ( $R$ -value) ~0.9999.
  - Nice fitting
  - In this case, “maximum applied load” and “holding time” were used as model parameters. There were three different “maximum applied loads” and eight different “holding times”. The randomness of experimental parameters is average when compared with other case studies. Thus, fitting is nice, but average, when compared with other case studies.
  - Fitting is acceptable. Though MSE can be reduced by re-training the model with the same model parameters or using a different set of parameters, number of neurons in the hidden layer, training, validation, and testing set size also affect the model accuracy and can be changed.
- Case study 2.2.2. Al-5CNT coating [24,25] in Appendix A:
  - MSE ~1.0 and correlation ( $R$ -value) ~0.9999.
  - Best fitting among the three case studies.
  - In this case, “maximum applied load” and “loading rate” were used as model parameters. There were nine different “maximum applied loads” and nine different “loading rates”. The randomness of experimental parameters is best when compared with other case studies. Thus, fitting is best when compared with other case studies.
  - Fitting is acceptable. Retraining the model is not needed as error values are quite low. MSE can be reduced and a value of zero means “no error”. Since ANN models are prone to “overfitting”, one must avoid trying to reach a value of zero for MSE.
- Case study 2.2.3. spark-plasma-sintered hydroxyapatite (HA) sample [28] in Appendix A:
  - MSE ~10.0 and correlation ( $R$ -value) ~0.995.
  - Comparatively bad fitting when compared among the three case studies.
  - In this case, “maximum applied load” and “loading rate” are identical in all the cases. There is no randomness for these two model parameters. Thus, fitting is comparatively bad, when compared with other case studies. Fitting is acceptable as we are dealing with a noisy data set. There is scope of improvement in fitting and can be done by working on reducing MSE. However, the problem is lack of randomness of two of the experimental test parameters. Due to lack of randomness of experimental parameter, combined with noisy nature of the dataset, there will not be any significant improvement in the fitting. For this type of dataset, “overfitting” is a major concern.

In case study 2.2.2., the least number of experiments were performed, while error metrics for the AI-based model developed for case study 2.2.2. are comparatively better than the error metrics for the AI-based models developed for other case studies. For developing AI-based models, it is important that design space is uniformly distributed. A large number of support points are helpful in developing more accurate models. In case study 2.2.2., there are nine different values of “loading rate” and “maximum applied load”. Model parameters are more random when compared with other case studies. Thus, best fitting is demonstrated by AI-based model for case study (2.2.2.) as experimental parameters are well distributed (random) and best suited for the development of AI-based models.

Through this platform, a user can simulate large number of virtual nanoindentation tests. Once they are satisfied with the nature of experimental and simulated load-displacement curve, they can move ahead towards estimating nanomechanical properties from the load-displacement curves. In the future, we plan to include estimating nanomechanical properties in this software/GUI.

#### 4.2. AI-Based Models for AFM Indent Image

Regarding AFM image prediction from raw HDF (.hdf) data, this is one of the first works of its kind and we have not found any article to compare it with our current work. AFM image has been analyzed through data-driven techniques [20,22], but through our approach, we can predict an AFM image for new experimental parameters which has not been reported in any prior work. In this work, we have demonstrated that we can develop AI-based models for AFM image from HDF (.hdf) files generated by the machine. Through our platform, a user can view the indents in 2D and 3D. Additionally, they can also predict AFM image for new test parameters, and all this has been done using data from just 18 sets of AFM images in case study (2.2.1.).

AI-based models for other case studies were not shown due to inadequate number of AFM files in HDF format. As mentioned in Sections 1 and 2, AFM imaging is not always performed.

For developing AI-based models for AFM image prediction, we recommend performing scanning through AFM in a well-planned manner, that is, performing AFM imaging for all the indents in either FORWARD or REVERSE direction, or both directions as described in Section 2. A few well-planned experiments will be helpful in developing accurate AI-based models for AFM image like in case study 2.2.1.

#### 5. Conclusions

After performing a nanoindentation experiment, an experimentalist obtains a load-displacement curve and an AFM indent image while corresponding data are stored in files in text (.txt) and HDF (.hdf) format. Mechanical properties of a material are estimated through load-displacement curve, while material deformation is visualized through AFM image.

In the present work, we developed a computational platform/software for virtual simulation of nanoindentation experiment for new test conditions. Simulated/predicted test results are in the form of indentation depth/load-displacement curve and AFM indent image. Key points are as follows:

1. Predictive models were developed by application of concepts of artificial intelligence on raw data files generated by the nanoindentation machine (data are in .txt and .hdf format).
2. We have successfully tested our approach on three different coatings/materials. Predicted results have been presented in the form of a load-displacement curve and AFM indent image. Predicted results are in good agreement with the experimental findings.
3. In this work, we used data from 9, 10, and 24 indentation experiments. Properly planned, nine experiments can be used for analyzing nanomechanical properties of a material like Case study 2.2.2.: carbon nanotube reinforced aluminum composite (Al-5CNT) coating. Properly planned here means randomized experimental test conditions like “maximum applied load”, “holding time”, and “loading rate”. The more randomized the values of these test conditions, the better the prediction capability of the model.
4. Model development for indentation depth and AFM image is less than an hour. Model execution time for indentation depth is less than a minute, while for AFM image, it varies between 2–4 min. Thus, a user can develop AI-based models and perform a few simulations within an hour.
5. This work can be helpful for experimentalists in significantly reducing the number of experiments performed by them while characterizing a coating/material through nanoindentation.
6. With respect to predicting AFM image, this is a first work of its kind. Researchers have analyzed AFM images and correlated it with experiments [20,22], but this is the first work where a user can predict an AFM image as a function of experimental test parameters.
7. We can easily adapt our platform for incorporating additional test conditions/parameters.

8. If a user names the files as per our instructions, a user just needs to place all the files in a folder and can try our “DEFAULT” option. This makes our method extremely simple for experienced experimentalists who do not need or want to invest time in understanding artificial intelligence algorithms. This software (GUI/APP) can be used as a screening tool prior to performing experiments. We have added options for users with AI expertise too.
9. All of this work was performed on a laptop. Thus, a user can use their current computer/laptop for this work.

This software (GUI/APP) can be installed under “My Apps” and can be used as a MATLAB toolbox for MATLAB users on any operating system. We have also prepared a standalone software for Windows operating system for users without a MATLAB license. They can use our standalone software for free after installing the MATLAB Runtime, which is also free.

### 5.1. Recommendations/Suggestions

**AFM imaging:** For using this platform, a user needs to arrange an adequate number of files. In particular, there must be some randomness in experimental test conditions. For AFM imaging, we are dealing with a large amount of data extracted from image files in HDF format. We have demonstrated that 18 sets of experiments were adequate for case study 2.2.1. A user must perform AFM imaging in a well-planned manner. For accurate prediction of an AFM image, a user must use both “gradient” and “topography” profile of an indent. AFM imaging must be performed in either “FORWARD” or “REVERSE” direction or both.

For load-displacement curve prediction, this platform can be used for testing other types of materials. We have demonstrated performance of our platform on three different types of materials, thus a user can use this platform in their own work. The randomness of experimental test parameters can be helpful in developing accurate models using data from a limited number of nanoindentation test (for example case study 2.2.2.).

Thus, the randomness of experimental test parameters is extremely important for minimizing the number of experiments required for characterizing a material.

### 5.2. Future Work

In the future, we will work on developing a module for estimation of mechanical properties like hardness, Young’s modulus, strain-rate sensitivity, and activation volume.

## 6. Patents

We are currently working on patenting our software and are also in direct communication with a company interested in it.

**Supplementary Materials:** Supplementary file and video abstract contains instructions on using this software. The following are available online at <https://www.mdpi.com/2079-6412/11/3/299/s1>.

**Author Contributions:** Conceptualization, R.J. and A.A.; data curation, R.J.; formal analysis, R.J.; funding acquisition, A.A.; investigation, R.J.; methodology, R.J. and A.A.; project administration, A.A.; resources, A.A.; software, R.J.; supervision, A.A.; validation, R.J.; visualization, R.J.; writing—original draft, R.J.; writing—review & editing, R.J. and A.A. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The datasets generated during and/or analyzed during the current study are not publicly available but are available from the corresponding author on reasonable request. We plan to include the raw data files as case studies along with the software.



**Conflicts of Interest:** The authors declare no conflict of interest.

## Appendix A

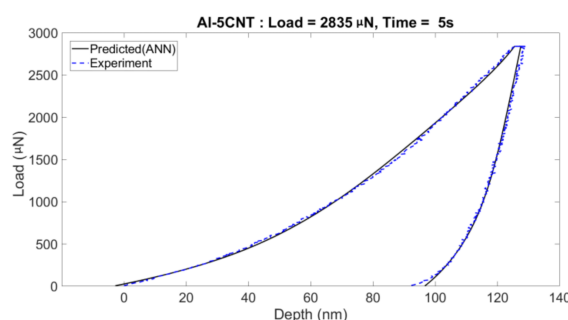
Case Study (2.2.2.) Prediction of indentation “depth” for carbon nanotube reinforced aluminum composite (Al-5CNT) coating [24,25]

For carbon nanotube reinforced aluminum composite (Al-5CNT) coating, holding time was fixed at 5 s for all experiments. In this case study, maximum applied load was varied. Loading rate was also varied for each value of maximum applied load.

We identified four input parameters to affect indentation depth (“depth” from text file) of material in this case study. These parameters are user defined “maximum applied load” and “loading rate” and two parameters (“load” and “time”) from the text file generated by the machine.

Thus, we developed an ANN model for “depth” as a function of “maximum applied load”, “loading rate”, “load”, and “time”.

There were nine text files corresponding to nine indentation experiments. We used eight of these files during model development for indentation “depth”. One text file was kept for comparing the nature of the load-displacement curve generated by the machine and the one generated by the ANN model for indentation “depth”. Figure A1 shows the comparison of these load-displacement curves, one generated from the experiments (text file) and one generated through ANN model for indentation “depth”.



**Figure A1.** Comparison: experiment vs. predicted for maximum load of 2835  $\mu\text{N}$ , loading rate of 283  $\mu\text{N/s}$ , and holding time of 5 s for carbon nanotube reinforced aluminum composite (Al-5CNT) coating.

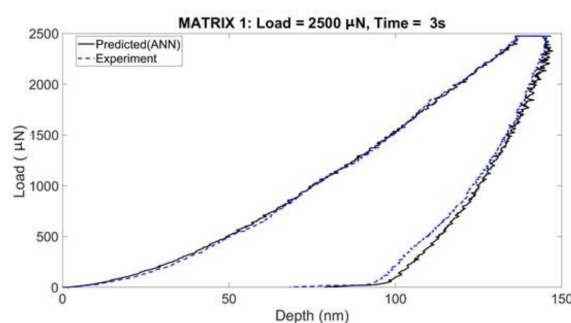
Case Study (2.2.3.): Prediction of indentation “depth” for spark-plasma-sintered hydroxyapatite (HA) sample [28].

In this case, 10 nanoindentation tests were performed on 2 different matrices. In each of these cases, “maximum applied load” and “holding time” were fixed.

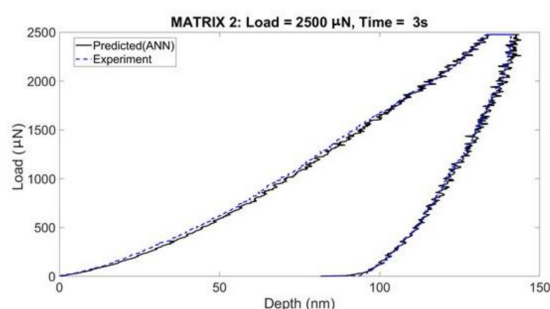
We developed an ANN model for “depth” as a function of “maximum applied load”, “holding time”, “load”, and “time”.

We had 10 text files corresponding to 10 indents for each matrix. Nine files were used during model development, and one file was kept for comparison for each of these matrixes. Figures A2 and A3 show the load-displacement curve comparison between experiments and ANN model for coatings obtained through spark-plasma-sintered hydroxyapatite (HA).

In this case study, the dataset was very noisy as can be seen in Figures A2 and A3. ANN models for indentation depth are able to capture the trend of the load-displacement curve. There is some error mainly because test parameters are constant in all the cases. Thus, a user must use data where experimental test conditions vary and avoid using lots of constant values.



**Figure A2.** Comparison: experiment vs. predicted for load of 2500  $\mu\text{N}$  and holding time of 3 s for spark-plasma-sintered hydroxyapatite (SPS-HA) for Matrix-1.



**Figure A3.** Comparison: experiment vs. predicted for load of 2500  $\mu\text{N}$  and holding time of 3 s for SPS-HA for Matrix-2.

An experimentalist checks the nature of the generated curve prior to considering an indentation experiment for further study materials response or estimate mechanical properties. Thus, our approach will be helpful for screening of user defined load and time variation for any specific maximum applicable load, holding time, and loading rate.

## References

1. Oliver, W.; Pharr, G. An improved technique for determining hardness and elastic modulus using load and displacement sensing indentation experiments. *J. Mater. Res.* **1992**, *7*, 1564–1583. [\[CrossRef\]](#)
2. Chu, Q.; Xu, S.; Zhu, X.; Zhu, Z.; Zhang, H.; Bai, R.; Lei, Z.; Yan, C. Effects of testing conditions on the deformation behaviour of a Ti-based bulk metallic glass. *Mater. Sci. Eng. A* **2019**, *766*, 138404. [\[CrossRef\]](#)
3. Liu, H.; Wei, Y.; Liang, L.; Wang, Y.; Song, J.; Long, H.; Liu, Y. Microstructure observation and nanoindentation size effect characterization for micron-/nano-grain TBCs. *Coatings* **2020**, *10*, 345. [\[CrossRef\]](#)
4. Qian, L.; Zhao, H. Nanoindentation of soft biological materials. *Micromachines* **2018**, *9*, 654. [\[CrossRef\]](#) [\[PubMed\]](#)
5. Staszczuk, A.; Sawicki, J.; Kołodziejczyk, Ł.; Lipa, S. Nanoindentation study of intermetallic particles in 2024 Aluminum alloy. *Coatings* **2020**, *10*, 846. [\[CrossRef\]](#)
6. Zhang, W.; Li, J.; Xing, Y.; Nie, X.; Lang, F.; Yang, S.; Hou, X.; Zhao, C. Experimental study on the thickness-dependent hardness of  $\text{SiO}_2$  thin films using nanoindentation. *Coatings* **2021**, *11*, 23. [\[CrossRef\]](#)
7. Xu, K.W.; Hou, G.L.; Hendrix, B.C.; He, J.W.; Sun, Y.; Zheng, S.; Bloyce, A.; Bell, T. Prediction of nanoindentation hardness profile from a load-displacement curve. *J. Mater. Res.* **1998**, *13*, 3519. [\[CrossRef\]](#)
8. SPM Modi and Measurement Methods. Available online: <https://www.dme-spm.com/spmmodi.html> (accessed on 20 February 2021).
9. Gibson, R.F. A review of recent research on nanoindentation of polymer composites and their constituents. *Compos. Sci. Technol.* **2014**, *105*, 51–65. [\[CrossRef\]](#)
10. Khezri, M.; Bradford, M.A.; Vrcelj, Z. Application of RKP-FSM in the buckling and free vibration analysis of thin plates with abrupt thickness changes and internal supports. *Int. J. Numer. Methods Eng.* **2015**, *104*, 125–156. [\[CrossRef\]](#)
11. Khosravania, M.R.; Nasiri, S.; Weinber, K. Prediction of fracture in sandwich-structured composite joints using CBR approach. *Procedia Struct. Integr.* **2018**, *13*, 168–173. [\[CrossRef\]](#)
12. Hu, J.; Sun, S.; Zhang, W.; Peng, G.; Han, S.; Sun, X.; Zhang, Y.; Zhang, T. Softening behaviors of severely deformed Zn alloy studied by the nanoindentation. *Coatings* **2020**, *10*, 803. [\[CrossRef\]](#)
13. Ye, D.; Wang, W.; Xu, Z.; Yin, C.; Zhou, H.; Li, Y. Prediction of thermal barrier coatings microstructural features based on support vector machine optimized by cuckoo search algorithm. *Coatings* **2020**, *10*, 704. [\[CrossRef\]](#)

14. Hufnagel, T.C.; Schuh, C.A.; Falk, M.L. Deformation of metallic glasses: Recent developments in theory, simulations, and experiments. *Acta Mater.* **2016**, *109*, 375–393. [CrossRef]
15. Horstemeyer, M.F. *Integrated Computational Materials Engineering (ICME) for Metals: Using Multiscale Modeling to Invigorate Engineering Design with Science*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2012.
16. Lu, L.; Dao, M.; Kumar, P.; Ramamurty, U.; Karniadakis, G.E.; Suresh, S. Extraction of mechanical properties of materials through deep learning from instrumented indentation. *PNAS USA* **2020**, *117*, 7052–7062. [CrossRef] [PubMed]
17. Koumoulos, E.; Konstantopoulos, G.; Charitidis, C. Applying machine learning to nanoindentation data of (Nano-) enhanced composites. *Fibers* **2020**, *8*, 3. [CrossRef]
18. Lee, H.; Huen, W.Y.; Vimonsatit, V.; Mendis, P. An Investigation of nanomechanical properties of materials using nanoindentation and artificial neural network. *Sci. Rep.* **2019**, *9*, 13189. [CrossRef]
19. Konstantopoulos, G.; Koumoulos, E.P.; Charitidis, C.A. Testing novel portland cement formulations with carbon nanotubes and intrinsic properties revelation: Nanoindentation analysis with machine learning on microstructure identification. *Nanomaterials* **2020**, *10*, 645. [CrossRef] [PubMed]
20. Germanicus, R.C.; Mercier, D.; Agrebi, F.; Fèbvre, M.; Mariolle, D.; Descamps, P.; Leclère, P. Quantitative mapping of high modulus materials at the nanoscale: Comparative study between atomic force microscopy and nanoindentation. *J. Microsc.* **2020**, *280*, 51–62. [CrossRef] [PubMed]
21. Becker, B.; Hintsala, E.; Stadnick, B.; Stauffer, D.; Hangen, U. Evaluating Uncertainty in Clustering of Nanoindentation Mapping Data, TMS Virtual 2021. In Proceedings of the Symposium “AI/Data informatics: Design of Structural Materials—Poster Session”, Pittsburgh, PA, USA, 15–18 March 2021.
22. Hintsala, E.D.; Hangen, U.; Stauffer, D.D. High-throughput nanoindentation for statistical and spatial property determination. *JOM* **2018**, *70*, 494–503. [CrossRef]
23. Karimzadeh, A.R.; Koloor, S.S.; Ayatollahi, M.R.; Bushroa, A.R.; Yahya, M.Y. Assessment of nano-indentation method in mechanical characterization of heterogeneous nanocomposite materials using experimental and computational approaches. *Sci. Rep.* **2019**, *9*, 15763. [CrossRef]
24. Chen, Y.; Bakshi, S.R.; Agarwal, A. Correlation between nanoindentation and nanoscratch properties of carbon nanotube reinforced aluminum composite. *Surf. Coat. Tech.* **2010**, *204*, 2709–2715. [CrossRef]
25. Bakshi, S.R.; Lahiri, D.; Patel, R.R.; Agarwal, A. Nanoscratch behavior of carbon nanotube reinforced aluminum composite coatings. *Thin Solid Films* **2010**, *58*, 1703–1711. [CrossRef]
26. Pitchuka, S.B.; Lahiri, D.; Sundararajan, G.; Agarwal, A. Scratch induced deformation behavior of cold sprayed aluminum amorphous/nanocrystalline coatings at multiple load scales. *J. Therm. Spray Technol.* **2014**, *23*, 502–513. [CrossRef]
27. Pitchuka, S.B.; Jha, R.; Guzman, M.; Sundararajan, G.; Agarwal, A. Indentation creep behavior of cold sprayed aluminum amorphous/nano-crystalline coatings. *Mater. Sci. Eng. A* **2016**, *658*, 415–421.
28. Lahiri, D.; Singh, V.; Keshri, A.K.; Seal, S.; Agarwal, A. Carbon nanotube toughened hydroxyapatite by spark plasma sintering: Microstructural evolution and multi-scale tribological properties. *Carbon* **2010**, *48*, 3103–3120. [CrossRef]
29. Jha, R.; Dulikravich, G.S. Discovery of new Ti-based alloys aimed at avoiding/minimizing formation of  $\alpha''$  and  $\omega$ -phase using CALPHAD and artificial intelligence. *Metals* **2021**, *11*, 15. [CrossRef]
30. Jha, R.; Chakraborti, N.; Diercks, D.R.; Stebner, A.P.; Ciobanu, C.V. Combined machine learning and CALPHAD approach for discovering processing-structure relationships in soft magnetic alloys. *Comput. Mater. Sci.* **2018**, *150*, 202–211. [CrossRef]
31. *MATLAB and Statistics Toolbox Release 2019b*; The MathWorks, Inc.: Natick, MA, USA, 2019.
32. The HDF Group. Available online: <https://www.hdfgroup.org/about-us/> (accessed on 26 September 2020).
33. *MATLAB and Deep Learning Toolbox™ (Formerly Neural Network Toolbox™) Release 2019b*; The MathWorks, Inc.: Natick, MA, USA, 2019.
34. TI 950 TriboIndenter User Manual. Available online: <https://engineering.unl.edu/downloads/files/UserManual-HystrionTriboIndenter-sm.pdf> (accessed on 26 September 2020).