Journal of Engineering Design

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/cjen20

Image regression-based digital qualification for simulation-driven design processes, case study on curtain airbag

Mohammad Arjomandi Rad, Mirza Cenanovic & Kent Salomonsson

To cite this article: Mohammad Arjomandi Rad, Mirza Cenanovic & Kent Salomonsson (2023) Image regression-based digital qualification for simulation-driven design processes, case study on curtain airbag, Journal of Engineering Design, 34:1, 1-22, DOI: 10.1080/09544828.2022.2164440

To link to this article: https://doi.org/10.1080/09544828.2022.2164440

© 2023 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group

Published online: 19 Jan 2023.

Submit your article to this journal

Article views: 302

View related articles

View Crossmark data

Citing articles: 1 View citing articles

Full Terms & Conditions of access and use can be found at
https://www.tandfonline.com/action/journalInformation?journalCode=cjen20

JOURNAL OF ENGINEERING DESIGN
2023, VOL. 34, NO. 1, 1–22
https://doi.org/10.1080/09544828.2022.2164440

Taylor & Francis
Taylor & Francis Group

OPEN ACCESS Check for updates

Image regression-based digital qualification for simulation-driven design processes, case study on curtain airbag

Mohammad Arjomandi Rad[a], Mirza Cenanovic[b] and Kent Salomonsson[c]

[a]Department of Industrial and Materials Science, Chalmers University of Technology, Göteborg, Sweden; [b]Department of Product Development, Production and Design, Jönköping University, Jönköping, Sweden; [c]School of Engineering Science, University of Skövde, Skövde, Sweden

ABSTRACT

Today digital qualification tools are part of many design processes that make them dependent on long and expensive simulations, leading to limited ability in exploring design alternatives. Conventional surrogate modelling techniques depend on the parametric models and come short in addressing radical design changes. Existing data-driven models lack the ability in dealing with the geometrical complexities. Thus, to address the resulting long development lead time problem in the product development processes and to enable parameter-independent surrogate modelling, this paper proposes a method to use images as input for design evaluation. Using a case study on the curtain airbag design process, a database consisting of 60,000 configurations has been created and labelled using a method based on dynamic relaxation instead of finite element methods. The database is made available online for research benchmark purposes. A convolutional neural network with multiple layers is employed to map the input images to the simulation output. It was concluded that the showcased data-driven method could reduce digital testing and qualification time significantly and contribute to real-time analysis in product development. Designers can utilise images of geometrical information to build real-time prediction models with acceptable accuracy in the early conceptual phases for design space exploration purposes.

ARTICLE HISTORY
Received 20 October 2022
Accepted 29 December 2022

KEYWORDS
Product development; image regression; dynamic relaxation; convolutional neural networks; data-driven design;

1. Introduction

Today many products are challenged by having transdisciplinary, iterative, and simulation-driven design processes. Fluctuating requirements lead the way to iterative solution finding (André 2017) in top-down development processes. Throughout the development process, a variety of models within different levels of description or granularity (Maier, Eckert, and John Clarkson 2019) are utilised. This utilisation dictates considerable design iterations as the model maturity level increases. Moreover, back-and-forth work between engineering teams with different specialists from various disciplines adds to design iterations and

CONTACT Mohammad Arjomandi Rad radmo@chalmers.se

© 2023 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group
This is an Open Access article distributed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives License (http://creativecommons.org/licenses/by-nc-nd/4.0/), which permits non-commercial re-use, distribution, and reproduction in any medium, provided the original work is properly cited, and is not altered, transformed, or built upon in any way.

increases development lead time (Arjomandi Rad 2020). Iterations are tied up with the nature of the development and therefore it is not desired to be reduced or eliminated. However, being able to qualify design ideas in real-time through an iterative design process, can give many industries an advantage in their product development.

During the design processes, simulations have been mainly used in the digital qualification phase to test the performance of a pre-built model. To this end, a wide range of simulations such as rigid body dynamics, finite elements/differences, computational fluid dynamics (CFD), discrete events, and so on are used. However, with so many sources for design iterations, it is not surprising that many companies, on top of well-known and commercially developed tools, attempt to develop in-house qualification tools and accelerate their design evaluation process. Figure 1 shows a process model for our studied design process that is challenged in this noted way. As depicted in the picture, during the development process series of simulations as an example of digital qualification are performed sequentially.

In this process, failing at each point takes the design back to the beginning point where all requirements need to be tested again. In such an iterative and simulation-driven design process, once everything is checked within the digital realm, physical testing and qualification begin to verify the design before lineup for the production. This creates additional iterations in case the design fails in the physical qualification phase, which requires all digital qualifications to be performed from the start.

One area dealing with iterative design processes that have offered many solutions to this problem is Design Automation. This field essentially aims to automate the activities that are carried out during an engineering design process (Johansson 2011). Not only it entails connecting and running engineering supports with various goals by computer codes but also goes beyond and covers a wide range of preparatory tasks necessary for running those scripts that range from digitising tacit knowledge in spreadsheets to parameterising, annotating, or even recently extending digital model constituents (Johansson 2014; Heikkinen 2021). Automation in the design process substitutes computers for human labour to achieve faster design iterations and therefore shows a greater benefit when human labour is an issue. However, as the illustrated challenging design process in Figure 1 shows, the majority of the time in the design loops is spent not by humans but by computers themselves for reading, processing, and writing the information. As an instance of processing

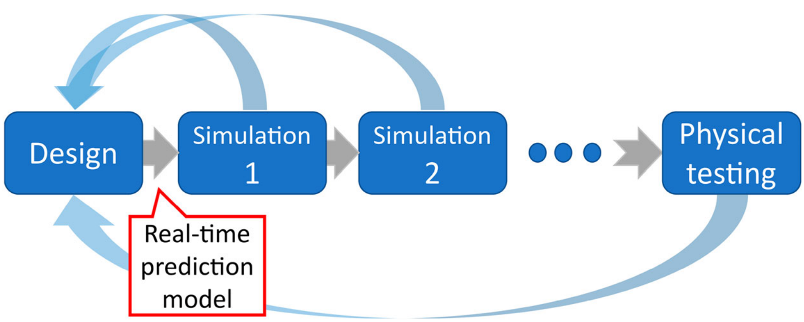


Figure 1. An iterative and simulation-driven design process.

the data, solving finite element simulations for a complex product can take weeks, and performing them even in an automated way by computers does not fulfil the solution criteria for fast design space exploration.

To Deal with iterative design, some studies have proposed Reasoning support for predicting requirements change and propagation, based on metrics that quantify requirements volatility behaviour (Hein et al. 2022). Other attempts to use network-oriented solutions to address changing requirements and predict their propagation exists (Dong et al. 2022). However, in most cases, these frameworks rely on PLM software processes or similar change management systems to get dependency relationship simulations.

Another area of research dealing with the problem at hand is the meta or surrogate modelling methods such as regression modelling, Grey relational analysis, Inductive learning, Group method of data handling, Kriging, etc., which have been used for decades to replace expensive computer analyses and cut down computational lead time (Simpson et al. 2001; Wang and Shan 2007). By cutting down computational time, these methods contribute to a lower development lead time which frees up resources and makes the design process more efficient. Additionally, the ability of these methods in capturing tacit knowledge in a company or bringing downstream production knowledge to the early conceptual phases and their response time after being created makes them highly interesting for addressing iterative design process issues.

Surrogate models are criticised to be narrow in the sense that they are specifically developed for the computational analysis stage of the development process and are dependent on previously acquired data to be able to function effectively (Fuhg 2019; Nunez et al. 2012). To overcome such issues, designers increase the design dimension, which leads to so-called high dimensional, expensive, and black-box (HEB) problems (Shan and Wang 2010) that require more sophisticated methods than straightforward statistics.

From a wider perspective, and looking out of the product development box, Artificial Intelligence (AI) which is mainly used to build surrogate models in engineering research has seen periods of reduced interest, called 'AI winters' (Crevier 1993) induced by hardware and software limitations. Today the latest wave of AI research is derived from bigger databases and deeper networks. Global AI competitions like *ImageNet* helped to increase the accuracy of the classification benchmark problems up to 97% which is essentially better than human performance (Soo Ko 2022; Panchal et al. 2019). Similar efforts in mechanical engineering to build annotated large-Scale 3D object databases have surged recently with the same hopes for increasing the availability of engineering benchmarks and the chance for higher precision for networks in future studies (Kim et al. 2020; Chang et al. 2015; Koch et al. 2019). The question if deep learning can address high dimensionality has been raised by other researchers (Li, Wang, and Liu 2017) but remains for discussion.

One well-established deep learning method associated with the latest AI waves is *Image Regression* in the computer vision field. The characteristic of this category as shown in Figure 2 are the input as images and continuous space as output. Age estimation and house price estimation are among the well-known problems for benchmarking developed algorithms. Using these problems many researchers try to estimate house prices from visual features from house photographs (Ahmed and Moustafa 2016) and also human age estimation from face portraits (Angulu, Tapamo, and Adewumi 2018). Examples of image regression in engineering problems exist that in many cases do not use such a convention to define their work (see Section 2.3). However, to exploit potential customised AI

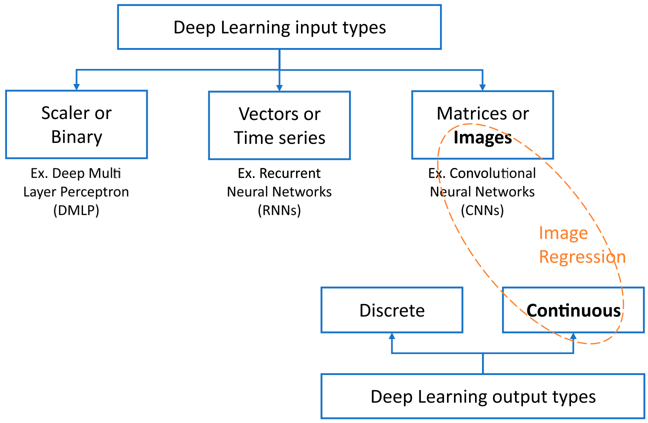


Figure 2. Input–output of deep learning methods with some examples.

algorithms in image regression and pave the way for more customisation on engineering databases this convention can be helpful (Panchal et al. 2019) and therefore is chosen in this paper.

The main purpose of this paper is to propose a prediction tool that can foretell simulation results while the design is being performed. The illustrated design process in Figure 1 shows such a hypothetical solution as a real-time prediction model right after the design process, shown in the picture by the red box. There are several considerations with regard to this hypothetical solution. One is the speed of prediction and the other is how much it can integrate with the design environment. Time is one of the main elements in the usability of such a tool in the design departments. A shorter time means faster design iterations between design tools and design teams in the industry. Moreover, the accessibility and usability of such a tool is equally important. Being able to be implemented as an add-on in the design environment will allow a wider audience to access an evaluation tool in the design process. It also helps to analyse more complex design variants within a short computational time and also makes it possible to navigate and assess the design space faster.

This paper contributes to the literature by showing how sophisticated image regression methods can be used by simply inputting screenshots from Computer-aided design (CAD) files as input for training. The accessibility of acquiring such screenshots in a CAD environment and the possibility of using noiseless pictures (not distorted and not blurred) is the advantage of the proposed framework in this paper. To the best of the authors' knowledge, this is the first database that utilises screenshots of the design in the CAD environment to produce input images for such purposes. The second contribution is providing a showcase for using dynamic relaxation as a means for labelling the database which conventionally counts as a drawback for applying surrogate models in the literature. Lastly, the image regression literature can benefit from engineering databases such as the one provided in this paper. This kind of database can be used for testing and benchmarking purposes in optimising image regression methods in engineering problems. Having engineering-based databases published will enable the developers to tailor algorithms for engineering-based problems.

This paper provides a database of curtain airbags and also presents a framework on how such databases can be built and utilised in the design processes as a real-time prediction tool. A critical review of the latest surrogate modelling techniques is presented in the second section. The third section describes the studied case which is the volume simulations of curtain airbags as an example of the noted design processes. Finite element simulations of the airbag are presented concisely in this section. In the fourth section, Dynamic Relaxation (DR) as a method is presented and used to acquire volumes of 60,000 CAD models with different geometrical configurations. In the next section, screenshots of each CAD model in the sketch environment are used to create input images for a database. The database is then verified by 100 randomly created design cases, evaluated by the finite element method (FEM). In section six, the verified database is used to train and test a CNN model as a real-time prediction tool. The final section discusses the contributions, and implications of the findings on how such models can help CAD designers know the consequence of their decisions in the early development phases.

2. Related research

The Literature indicates the most high-priority challenge for the yet-to-mature field of data-driven design is the lack of methodologies (Briard et al. 2023). Several literature reviews have attempted to point to hindrances along the way of using data science in engineering design (Chiarello, Belingheri, and Fantoni 2021; Jiao et al. 2021) and suggest several areas for future studies.

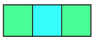
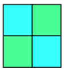
The parameterisation of CAD models is one of the well-known issues in applying surrogate models to design problems (Mohammad et al. 2022; Chiarello, Belingheri, and Fantoni 2021; Umetani and Bickel 2018). Parameterisation limits the designers to follow the same convention every time they need to design something, yet in real cases, the design might lose or take some geometric features. Therefore, parameterisation limits the creativity needed for the design work and results in a limited ability to explore the required design space. The number of the utilised parameters in a data-driven design approach affects the accuracy of the prediction model. Most of the conventional metamodelling approaches have been challenged by the curse of dimensionality which implies that the performance or accuracy of a system is reduced by an increased number of dimensions.

Conventional parametric model-based surrogate modelling techniques are a couple of decades old and essentially utilise relatively simple statistical methods normally with scalar inputs. Additionally, not all the surrogate modelling techniques are aiming for data-driven design-aiding solutions. The studies that are connected to pre-design and development work (such as market analysis or Kansei engineering) and also to post-design stages (such as production or after-market are excluded. As another criterion for reviewed papers, developing a design qualification/testing or predictive decision-making is used.

This section aims to present and categorise existing papers that are using recently developed data-driven algorithms to build real-time analysis or present similar surrogate/metamodels to cut computational and design lead time. Interestingly, all the reviewed publications were published in the last five years which was not an intentional limitation in the performed search strings.

The review reveals that most of the selected geometries in these papers are of very simple shapes. It can be argued that these novel data-driven models are not developed enough

Table 1. Reviewed papers based on their input type.

Input types		References
	Scalars or Binary	Secco and Silva de Mattos (2017) and most of the papers in the Simpson et al., Wang and Shan
	Vectors or Time series	Cao et al. (2021), Deshpande, Lengiewicz, and Bordas (2021), Cunningham, Simpson, and Tucker (2019), Belaid, Rabus, and Krestel (2021), and Vurtur Badarinath, Chierichetti, and Davoudi Kakhki (2021)
	Images or Matrices	Guo, Li, and Iorio (2016), Wang et al. (2018), Umetani and Bickel (2018), Khadilkar, Wang, and Rai (2019), Ferreira and Bell (2020), Messner (2020), Nie, Jiang, and Burak Kara (2020), Zhao et al. (2021), Yoo et al. (2021), Toro, Wiberg, and Tarkian (2022), and Du et al. (2022)

and lack the ability to deal with complex geometries in real-case scenarios. Moreover, labelling these datasets in all of the reviewed cased have performed by costly simulations such as finite element analysis. Moreover, a publicly available and well-developed engineering image regression database is still missing in the literature. To easily navigate between the different papers and know which ones can be categorised as image regression, we categorise the reviewed papers based on the type of input they are using as this is one of the advantages of recent algorithms. The categorisation sheds light on the wide range of new possibilities that new development in this area has awakened. Table 1 summarises the papers reviewed in this section.

2.1. *Scalars or binary parameters as input*

A large number of papers are those that use scalar parameters as inputs. This category has existed for several decades and it mostly included the type of HEB problems described earlier. For instance, Secco and Mattos created a database of 100,000 aerodynamic cases to predict CAE outputs such as lift and drag coefficient for wing-fuselage configurations (Secco and Silva de Mattos 2017). In total, 40 input variables were used. Several multi-layer feed-forward Artificial Neural Network (ANN) was trained using a scaled conjugate gradient algorithm. It was concluded that it is possible to set up an ANN to substitute an expensive computational analysis. However, the drawback with such databases is the mentioned parameterisation and dimensionality which hinders the development of this class of metamodels.

2.2. *Vectors or time series as an input*

In this category, the inputs are vectors or time series and represent geometrical information or Finite Element (FE) problem definition. The motivation is usually supported by the fact that for complex geometries, the bottleneck for surrogate modelling is parameterisation which leads to the loss of a lot of detailed geometrical information. For example, Cao et al. used geometric information of a surface mesh with Graph Neural Networks (GNN) to apply the surrogate modelling method (Cao et al. 2021). In their method, control points of the geometry are used to generate non-uniform rational B-spline (NURBS) surfaces, which are later used to get the surface mesh as the input of the surrogate model. The GNN was able to extract geometric information from the surface mesh of the designs automatically and also predict the output of the model which was the fluid domain on a special kind of turbine exhaust system.

As another example of using geometrical information data, researchers used nodal forces as input and proposed (Deshpande, Lengiewicz, and Bordas 2021) a probabilistic approach for predicting a 2D/3D beam displacement in real-time. In the study, the input is the load vector of each node and the output is the finite element response which is a vector of displacements for the same node. The method is still young and does not extrapolate well when the force is applied to those nodes that were not part of the training process.

Apart from the mesh and nodal forces, point cloud representation is another form of geometrical information that has also been used as an input. For instance, Cunningham et al. used two databases consisting of 2500 point clouds that belonged to 250 watercraft and aircraft 3D models to predict their CFD simulation results such as drag and lift coefficients (Cunningham, Simpson, and Tucker 2019). They performed the prediction on several sets of surrogate models and their results show that non-neural network models (such as radial basis functions) can achieve comparable accuracy to the neural network models (such as the well-known ResNet architecture). They concluded that neural networks should not be treated as 'one size fits all' and every problem should be checked for the most efficient network.

Belaid et al. proposed CrashNet which is a deep neural network architecture to predict crash test outcomes in the automotive development process, a calculation that is traditionally performed by FE simulations (Belaid, Rabus, and Krestel 2021). This was achieved by formulating car crash events as a time series prediction and using multiple scalar features and the car's acceleration time series as input. The prediction output was a time series of the occupant's chest acceleration which is an injury severity metric. As another example, Badarinath et al. studied the potential of a surrogate modelling tool with real-time data as input for monitoring stress distribution over a beam and defined a customised maintenance schedule (Vurtur Badarinath, Chierichetti, and Davoudi Kakhki 2021). Three well-known methods namely Gradient boosting regression trees, random forest, and artificial neural networks are used to train on the acceleration of the beam on five specific reference positions that are later extrapolated and used to predict the acceleration on any other point on the geometry. Although the utilised geometry was simple, it demonstrates a successful health monitoring system on a time-varying mechanical system.

2.3. *Images or matrices as input*

For this group, the input is in the matrix form and holds grey scale pixel values. Generally, three matrices represent Red, Blue, and Green (RGB) values or some other form of geometrical information in form of matrices. Whereas when the problem is 3D the matrix changes to the tensor form. Although, geometry can be represented in multiple ways and some of these representation forms can be embodied in matrices. However, Guo et al. argue that not any kind of representation can be effective since its semantic meaning can vary (Guo, Li, and Iorio 2016). One effective example demonstrated in their study is called Signed Distance Function (SDF) which can provide a universal representation for different geometrical shapes. Guo et al. successfully show how mathematical concepts like SDF sampled on a Cartesian grid are more effective for geometrical shape representation in CNNs.

To avoid parameterisation, a technique that can predict the fluid flow shape around a given three-dimensional object interactively is presented (Umetani and Bickel 2018). A database of different objects is gathered from the ShapeNet repository and used in CFD analysis to label the input data as PolyCube 3D projection of the objects with their aerodynamic performances such as velocity and pressure.

A prediction model is proposed (Wang et al. 2018) for the separation stress evaluation of a 3D printed part during the pull-up process. The input to the database was images of the shapes of the contact zones which are created by n-fold symmetric shape functions. The output was discrete locations of stress as a square grid. More 3D printing application examples exist (Khadilkar, Wang, and Rai 2019) where deep learning has been used to predict stress in the 3D printed parts. The input images were created from STL files with available data augmentation techniques. The distribution of stress on layers (i.e. 2D stress grid for each layer as the output) is calculated from finite element analyses.

Some studies (Ferreira and Bell 2020) utilise images as input to predict the output of onboard sensors in experimental aerodynamic tests. Creating a training database like that from real-life images can be time-consuming. Moreover, the sensors that are attached to the plane, due to the physics of the test, can introduce misleading measurements and increase the error in assumptions during the design phase. Therefore, in this study images of a generic aeroplane model together with perspective views of CAD models are used to build a database to predict the outputs such as angular measurements in wind tunnels (pitch, roll, and yaw) in the aeroplane design process. Two sets of CNN were used including the well-known architecture VGG-16. The reported error values less than 0.5° for each angle indicate promising predictive performance.

Messner built a surrogate model with CNN to evaluate the mechanical properties of square-symmetric, periodic composite structures (Messner 2020). The problem geometry in this study is discretised into unit cell square regions like pixels which are used as input. The drawback of this method is the generalisability of the method since this special discretisation might not work for every material or geometry. Nei et al. used images of a 2D linear cantilevered beam as input and solved 120,960 different configurations of the geometry with an FE model and trained a CNN that can predict the stress field as a picture (Nie, Jiang, and Burak Kara 2020). This end-to-end surrogate modelling was limited by having simple geometry, boundary, and load conditions. As a heat transfer application, other researchers (Zhao et al. 2021) developed a deep learning-based surrogate model with data augmentation and transfer learning that has been used to make a quick and high-precision temperature field prediction for some heat sources. The major limitations of these studies are that either the domain shape or the finite element simulation was too simple which can be more complicated in real-life problems. As another end-to-end study, Yoo et al. introduced a CAD/CAE integrated deep learning framework for generative design and studied the application of this system on an automotive road wheel design process (Yoo et al. 2021).

More recently researchers (Toro, Wiberg, and Tarkian 2022) have proposed using images of b-pillar panels from different vehicle models as input and locators of fixture layout as three locating points (3-2-1 principle) as output within a deep learning algorithm. The framework helps with the design of fixture layouts in the sheet metal industry. Another recently purposed real-time evaluation framework for modelling in the field of aerofoil design utilised deep learning (Du et al. 2022). In this study, the input was a combination of both images of aerofoil shape and operating parameters which were mapped to aerodynamic performance namely, the flow field. Apart from having relatively simple geometry which seems to be the limitation of the many studies in the category, the proposed framework reduced the time cost for aerofoil evaluation from 20 min with CFD to just 5 ms.

3. Studied prototype and finite element simulation

In this paper, the curtain airbag has been chosen as a case product. The curtain airbag design process is highly iterative and simulation-driven (Mohammad et al. 2022) and has several transdisciplinary requirements from different stakeholders. This product has a roughly two years development lead time where the major part of this time goes into dealing with iterative requirement satisfaction.

During the airbag design process and in the first step of the design assessment, the volume is controlled. This is because in the early stage of the design process the volume of the bag determines which inflator capacity should be used for a particular design. Since inflators are big factors in the final cost of the airbags, choosing the wrong inflator might lead to an overpriced quotation or even an unwanted deployment. As for the rest of the design process, other simulation models are used for other requirements.

To study this design process, a prototype geometry that represents the important aspects of real curtain airbags and its complexities, is designed and fully defined in *CATIA*. All 14 CAD parameters used to create different configurations from the geometry are shown in Figure 3.

These parameters are selected together with airbag designers in the case company. The selection criteria were to choose those parameters that are usually changed for producing the new shapes in real-case scenarios. Therefore, Some of the other parameters that do not follow this criterion are assumed to be constant. The selected parameters are varied between the minimum and maximum bounds to create different shapes. The bounds are

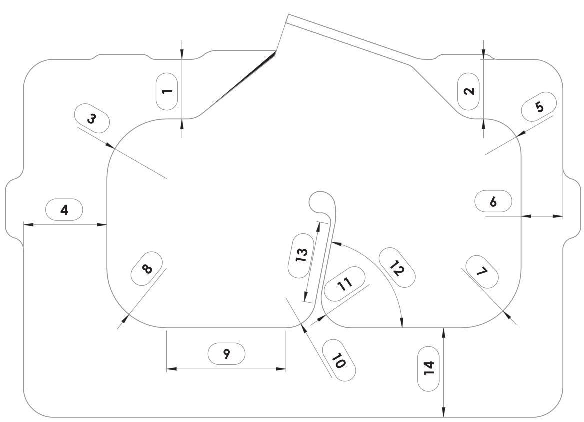


Figure 3. Selected parameters to create different configurations of the geometry.

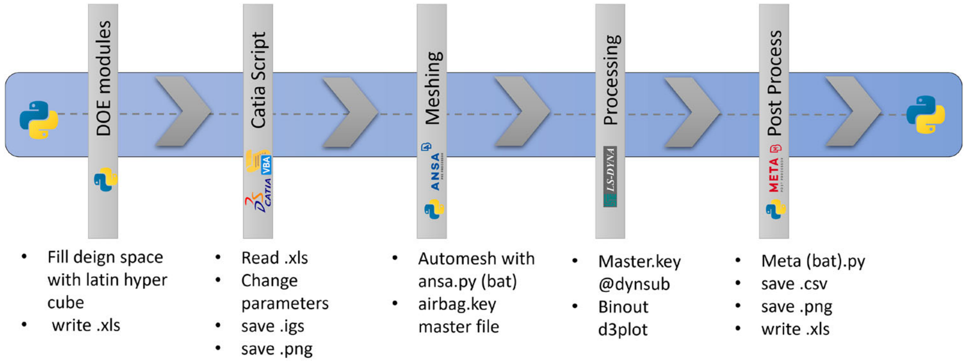


Figure 4. Design automation process used to create all the databases.

selected to make the parameters independent and prevent design failure in the realisation process.

To generate all the databases, a fully automated python script is used as a wrapper for executing different codes. A process view of this script is shown in Figure 4 which essentially connects different design tools together. It should be noted that some of these tools also use a special version of python as their scripting language in their local environment. As shown in the picture, a design of experiments *Latin Hypercube* python script is used on the mentioned 14 parameters to attain the required number of samples at each database within the mentioned bounds (min–max). In the next step, these design cases are generated using *CATIA* VBA script. At the same time and before saving the geometry, the script takes a screenshot from a defined angle and saves the image. The image file (as png) is used as input later and the geometry file (as igs) is used in the finite element analysis to calculate the volume after deployment. In the continue, the main python scripts run two additional scripts for meshing and processing through *ANSA* and *LS-DYNA*, respectively, as shown in the same figure.

For meshing the geometry, the auto mesh function is used and the generated key files are saved which are then passed into the processing step in LS-DYNA. For the finite element method (FEM), the Uniform Pressure technique (Hirth, Haufe, and Olovsson 2007) is used where the volume of the airbag is calculated by applying the Gaussian integral theorem. The method is known for stability, speed in run time, ease of implementation, and also to yield acceptable and accurate results. The output of this step was the d3plot files for each design case.

The main python code then executes a *META* post-processor script to read the d3plot files (simulation output). The volume is measured at a certain time (100 ms) as an ending criterion to resemble a car crash incident in a real-case scenario. During this time frame, the pressure in the airbag is raised to 40 kPa and kept constant.

To measure strain and stress rate, the mean integration point (IPT) effective strain and effective strain (von Mises) rates are selected in *LS-DYNA* which is illustrated in four different time frames in Figure 5. The figure demonstrates two hanging lug positions on the top of the bag and also how the stitched island inside the bag that is affecting the deployment.

For the current prototype geometry, the simulation run time is 10 to 15 min but for a real-size bag, it can take up to around 45–60 min depending on the size and complexity of

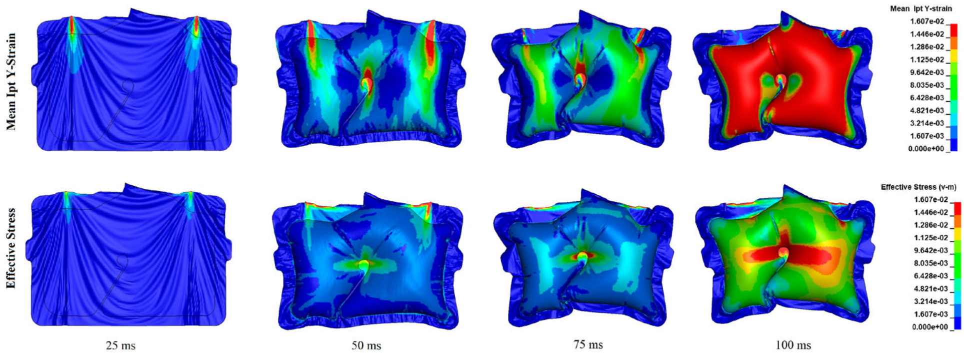


Figure 5. Example of stress and strain for an airbag design during deployment.

the bag in various classes (i.e. sedan, coupe, SUV, etc). On one hand, creating a big database with this method is not feasible because just 10 thousand simulation runs will take around 3 months for the small prototype used in this paper and 10–12 months for the real-size airbag used in industry. On the other hand, using such an automated script does not fulfil the industry's needs for envisioned real-time evaluations in the Introduction section. Therefore, to solve the issue a new framework that can label 60 thousand images in a reasonable time is introduced in the next section.

4. Computation of volume using dynamic relaxation

The bottleneck of applying machine learning-based surrogate modelling to engineering models has always been to build large amounts of labelled training data. This process can be prohibitively time-consuming if the computational models are too complex. Some simplifications of the models need to be made for the process to be feasible. Simplifications introduce additional errors which need to be taken into consideration and controlled.

A method for form-finding is Dynamic relaxation, which can be used to quickly compute volume for a large number of cases. Dynamic relaxation is an explicit method for static analysis of discretised structures. It has been used for modelling of, e.g. high pressure confined structures, thin pre-stressed fabric membranes, and for form-finding of civil and architectural structures such as hanging cables, chains, and domes. The method solves a static force-equilibrium problem by reformulating it as a dynamic problem and then minimising the total energy of the system by iteratively updating the mesh, such that the sum of the kinetic energies acting on all the nodes in the system is as low as possible, i.e. until the system reaches static equilibrium (Wong 2013). In the current paper, for implementing DR a component in Rhino ®-Grasshopper named 'Kangaroo' is used. Kangaroo is a live physics solver that is mostly used for interactive simulations, form-finding, optimisation, and constraint solving. We shall briefly discuss the differences between traditional computational models used for computing the volume of an airbag and the DR approach.

The LS-DYNA time integration method is stated as the semi-discretised system

$$\boldsymbol{M}\ddot{\boldsymbol{u}} + \boldsymbol{p} = \boldsymbol{f}$$

where $\boldsymbol{M}$ denotes the mass matrix, $\ddot{\boldsymbol{u}} = \frac{d^2\boldsymbol{u}}{dt^2}$ denotes the second time derivative of the displacement field, $\boldsymbol{p} = \boldsymbol{S}(\boldsymbol{u})\boldsymbol{u}$ denotes the internal force vector, with $\boldsymbol{S}$ denoting the stiffness matrix and $\boldsymbol{f}$ denotes the external and body loads.

This can be reformulated as

$$\boldsymbol{M}\ddot{\boldsymbol{u}} = \boldsymbol{r}(\boldsymbol{u})$$

where $\boldsymbol{r}(\boldsymbol{u})$ denotes the residual force field. For time step n a centre difference time integration leads to

$$\ddot{\boldsymbol{u}}_{n+1} = \boldsymbol{M}^{-1}\boldsymbol{r}(\boldsymbol{u}_n)$$

and in general, we have the displacement field in the explicit form

$$\boldsymbol{u}_{n+1} = \boldsymbol{M}^{-1}g(\boldsymbol{u}_n)$$

where $g(\boldsymbol{u}_n)$ denotes the coupled time discretisation terms of the dynamic equilibrium problem. The explicit time integration method is conditionally stable and there is a relation between the mass matrix and the critical time step, which needs to be controlled for the method to be stable. Additionally, the mass matrix is made diagonal for the method to truly be cheap, which is required since the time steps are short. Since the method is applied on shell or continuum elements, the spatial discretisation and subsequent FEM are quite computationally heavy compared to a simpler approach taken in dynamic relaxation.

In dynamic relaxation, the fictitious dynamic problem is stated as

$$\boldsymbol{M}\ddot{\boldsymbol{u}} + \boldsymbol{C}\dot{\boldsymbol{u}} + \boldsymbol{S}\boldsymbol{u} = \boldsymbol{f}$$

where $\boldsymbol{C}$ denotes the damping matrix. This can be rewritten as

$$\boldsymbol{M}\ddot{\boldsymbol{u}} + \boldsymbol{C}\dot{\boldsymbol{u}} = \boldsymbol{r}(\boldsymbol{u})$$

The time integration scheme is given by an e.g. a central differencing scheme resulting in the same form as seen in the case of LS-DYNA. The difference, however, is the spatial discretisation. The implementation of DR in Grasshopper uses simple trusses, or springs as elements, which are the simplest type of elements, with only stiffness as a 'material parameter'. Furthermore, the DR method uses fictitious time steps and a diagonalised fictitious mass matrix which, for the sake of stability, does not need to adhere to the same constraints as in the physically accurate method used by LS-DYNA. Thus the time steps, damping coefficient, and mass matrix are chosen in such a way as to minimise the number of iterations.

The discussion here on the differences between the methods is short and quite general since the implementational details of the DR in Grasshopper are not publicly known. However, Similarities and differences between DR and the Newton–Raphson method have been studied by Rombouts et al. where it is shown that the DR used in an implicit formulation is a special case of Newton–Raphson (Rombouts et al. 2018). It is also shown that the tangential stiffness matrix used in Newton–Raphson can be seen as a well-chosen fictitious mass matrix. The authors conclude that the DR method is suitable for highly nonlinear responses, and it is relatively easy to implement.

The same geometry files (.igs) from the first 100-design database, created in the previous section, that are used in FE simulations are used one more time to compute the volume

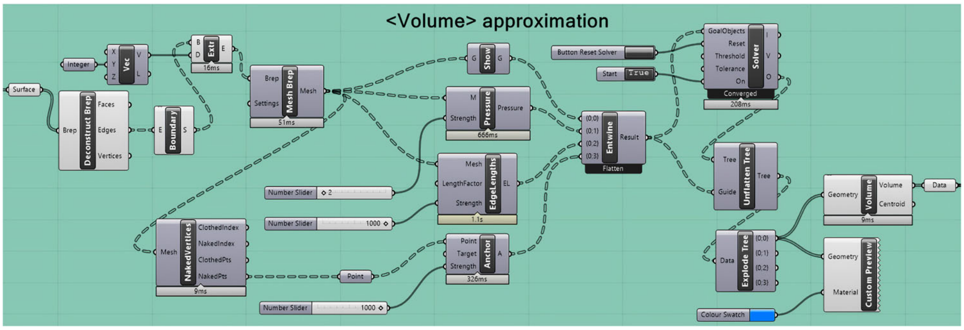


Figure 6. The process for Volume simulation in the Grasshopper.

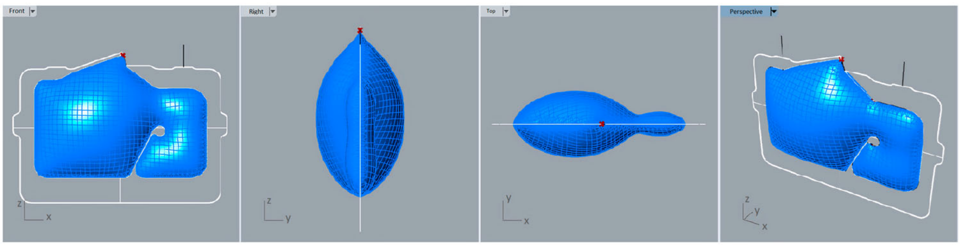


Figure 7. Volume simulation visualisation in the Rhino.

with the dynamic relaxation method by using the Kangaroo solver in Grasshopper. As illustrated in Figure 6 first all the geometries are imported from .igs files in Grasshopper and then decomposed into BREPS with a standard component. Then the BREPS are used to produce the mesh within another component. In the next stage, all the nodes on the edge of the geometry are anchored (clamped) and all the other middle nodes are subjected to internal pressure. The *Show* component is used for visualisation and the *edge length* as a margin which allows the code to identify the boundaries. Together, all four goals (see Figure 6, Show, Pressure, Edge length, and Anchor) are used as input for the Kangaroo solver. The rest of the components are for visualising the shape in a 3D geometry from which the volume is extracted and stored. The results are visualised with front, right, top, and perspective views captured from Rhino (see Figure 7).

To investigate how much DR answers are close to FE results, the calculated volumes from the two dynamic relations in this section are compared with the volumes that are calculated from finite elements for the same first 100-design database, performed in the previous section. In this way, the Pearson correlation method is used to analyse the correlation between these two methods over the same design cases. The results depicted in the middle step of Figure 8 show an accuracy of $R^2 = 0.9926$ between the output of the two methods. This relation between the two outputs proves that Dynamic Relaxation can be successfully employed for finding the volume of inflatable structures such as airbags. This correlation can also be employed to measure new samples' volume output. Simply by calculating DR volume output, the correlation can give the output for FE volume and vice versa.

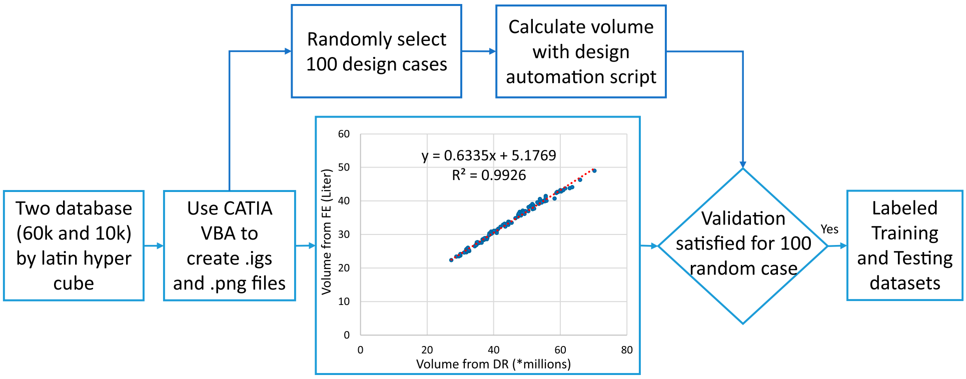


Figure 8. Labelling process using the correlation between Dynamic Relaxation (DR) and Finite Element (FE).

5. CAD-based database for image regression

This section presents the generation of the database consisting of 60,000 CAD models as an input, and also the calculation of volumes for each design as a label. Two initial scripts, namely Latin hypercube and CATIA VBA, in the described design automation code in the previous section are employed to create two databases with the size of 60,000 and 10,000 design cases (just the .png and .igs files). This process goes fast and does not take more than 2–3 h. The correlation mentioned in the previous section which is shown in Figure 8 is used to map the dynamic relaxation volume output to the FE volume output. This correlation proves to map DR results to FE output, which is possible by using the correlation that can be found from their comparison.

As shown in the figure, the outputs that have been derived by mapping dynamic relaxation to finite elements have been verified to make sure that they are accurate enough and are working not only on the mentioned database but also will work on a separate testing database as well. To do so, 100 design cases from the 60,000 images database are randomly chosen. Then the described design automation script in the previous section is used to measure the volume simulation of each model. The Pearson correlation method is used to find out the error between the output from the FE method and the output that was created by mapping DR. The error is reported to be under 1% which is considered an acceptable range.

By following such a process, the simulation run time was shortened from 10–15 min (for FE) to around 10–15 s (for DR), i.e. a reduction of 98%. More interestingly in Grasshopper, file handling, such as opening the file, reading, saving, and closing the file take more than the simulation itself which justifies why it is important to pursue real-time prediction models in a CAD environment. For this study, performing all volume calculations for 60,000 CAD files took around one week which could have taken more than one year in FE. Moreover, by taking steps further and making such analyses available in real-time in a CAD environment, designers can evaluate their decisions independently from discretisation or any file handling.

When each file was realised in CATIA just after saving the CAD file, a Visual Basic for Application (VBA) script was used to export screenshots of the geometry in the sketch environment and then save each image as a separate file. Furthermore, a python script was

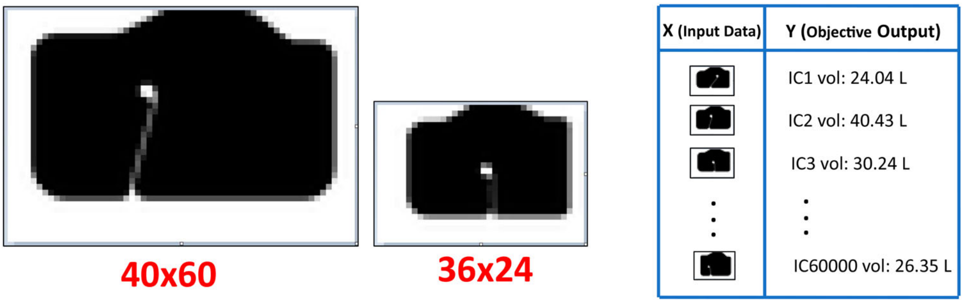


Figure 9. Examples from the database consist of 60,000 labelled images.

used to crop the screenshot files, remove the unnecessary areas and convert them into single-channel greyscale images. This was because it is sufficient if each pixel carries only information about the intensity and information from red, green, and blue (RGB) layers will not contribute to the learning process. At the end of this process, each image had a resolution of 1200×900 pixels.

Since screenshots are taken automatically from a predefined angle in CATIA, the geometry has similar positioning in all the images which is an interesting aspect associated with the proposed databases. This process makes sure the constant pixels fall on top of each other in every image which can help the learning process. This is usually not the case for other databases on image regression such as 'face to age' or 'house to price' databases. And these databases after being scrapped from the web, need to go through an enormous amount of pre-processing like cropping and editing.

The collected database is composed of 60,000 images (png files) in their original size and their associated labels. This database has been made publicly available in Kaggle (Arjomandi Rad 2022) by the authors for everyone so it can be used as a benchmark problem for algorithm improvement purposes. To enable the reuse of the data by other researchers, two databases with reduced sizes of 24×36 pixels and 40×60 pixels were also created within pickled and zipped (.pkl.gz) format and are made available publicly. This format allows for all the images and their associated labels to be stored in an n-dimensional python array. Figure 9 shows examples from the 24×36 and 40×60 pixels databases, respectively.

The figure shows that moving from a small size like 36×24 to 40×60 can have a big effect on the pixels that represent some of the small parameters like island angle or some of the radii in the corners. By trying several sizes, we conclude that 40×60 is a reasonable size and it can represent most of the changes in different geometries in the database.

6. A real-time prediction model

In this section, a convolutional neural network will be applied to the database that was built in the previous section to illustrate that CAD screenshots can be used to predict a simulation output in real-time. The network trained in this section, can be used as an estimator during the design phase and enable the designer to know the consequence of the decisions that are being made while working with the CAD sketch. For example, by defining a button in CATIA that runs a code in macro, we can execute a simple process that generates a screenshot from a certain view in CAD and use it as input to estimate the volume output for it. This

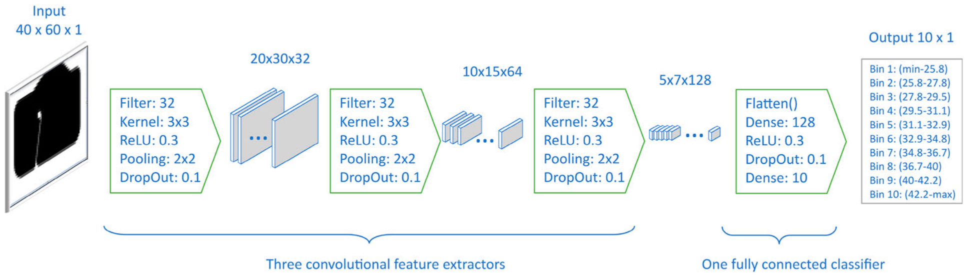


Figure 10. The architecture for utilised convolutional neural networks.

part is considered trivial and will not be discussed in the current paper. However, we do discuss how the engine behind the macro code or the simple architecture CNN architecture can be built.

Convolutional neural networks are developed to deal with images as their inputs. Among the reviewed papers in the second section, CNNs have been used successfully in a wide range of applications to map images to desired outputs. In the current paper, an implemented CNN is used in Tensorflow to build a class of customised architecture. Tensorflow is an open-source machine learning platform with flexible tools and a very large community that let researchers quickly develop and deploy various algorithms on different applications.

CNNs have two important functions inside, called convolution and pooling. Inside a convolution, several filters are used for scanning the input image which results in decreasing its size and increasing its dimension. Later the pooling is used to compress the images and cut down the size even more while keeping the dimension the same. This process is repeated several times with the aim of decreasing the image size and extracting features with different filters. At each repetition (i.e. layer) the images go through the ReLu activation before or after the pooling process. Moreover, to prevent the network's dependency on the training data (known also as memorising the data) and over-fitting, we use dropouts in every layer. In the last layer, the convoluted data gets flattened and then densified two times to change the dimensionality of the output (from proceeding convolutional layers) to match up in size with the values of the data in which the model is working.

Figure 10 shows the architecture of the utilised CNN in the current paper with three convolutional layers that compress and extract features from the input images and one fully connected flat layer that flattens, densifies, and connects finally to the output bins. In this implementation, the hyperparameters such as kernel size, alpha, pool size, activation, and padding type in the network are default values and the other parameters (such as dropout rate, input shape, and layer density) are chosen based on trial and error and best practices from literature.

To be able to use CNN on our database, the regression problem is changed to a classification problem in the current paper. To this end, all 60,000 numbers for volume are sorted and divided into 10 different bins each with 6000 samples. This is because the distribution of the volume between minimum and maximum was not equal and dividing them based on distance would make some of the bins have more samples than others. Therefore, this

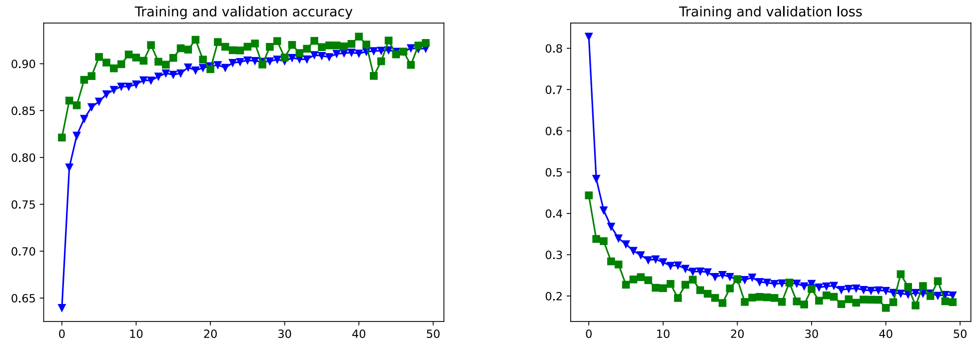


Figure 11. Accuracy (Left) and loss (Right) for the training and validation databases, respectively.

imbalance would have caused the network to give higher chances for some bins than others. To prevent this, the bins are divided based on the number of samples instead of their range.

To create the image regression model, the database of 60,000 images with a size of 40×60 pixels is divided into 5 sections. The model is trained using 4 sections of 48,000 training and one section of 12,000 evaluating sets. This model is then cross-validated for 5-Folds, using each time the same 4-1 section ratio. The training was performed using a batch size of 128 and epoch number of 50 and it took approximately 3 h on a normal core i7 desktop computer.

As the final step, the testing database with 10,000 new samples was used. As mentioned at the beginning of Section 5, the testing dataset was created separately to make sure the latent space includes entirely new shapes and differs from those which are used in the training and evaluating database. The testing showed roughly 89% accuracy and 0.26 loss which means that from 10,000 testing cases, 8943 cases were predicted correctly and 1057 cases were placed in the wrong bin. Results are shown in Figure 11. The testing is done fast under 1 min on the same computer.

In Figure 11, the loss value which is representing the summation of the errors in our model (calculated from the cost function) for each case in the testing database is shown on the right. The curve on the left shows the accuracy which is the percentage of the correct prediction and can only be applied to classification tasks. As can be inferred from the figure, the accuracy gap between training and validation converges to roughly 90%. Also in the loss graph, both training and validation are decreasing well and in a stable manner with an acceptable gap between them (known as the generalisation gap) with every epoch and converging to 0.2 which shows a successful result.

The presented training and testing process for a simple CNN architecture shows the model can predict the volume output with acceptable accuracy that is needed for early design phases. This CNN model is not optimised and there is definitely room for a couple of percent improvement by small changes, but this is left for future studies. It is also worth mentioning that during many iterations in the early development phases of a product, having a fast requirement checker that uses the design shape to predict a simulation output (at least with 90% accuracy) will accelerate the design process. Neural network-based models are essentially approximation models and therefore are not expected to give us a 100% accurate answer in a requirement evaluation process.

7. Discussion

Existing digital qualification and testing entail going through iterations of simulation tools. This process is often sequential and any small change in the design will require rechecking almost all other requirements. This issue hinders rapid design space exploration and also evaluation of radical ideas during product development processes. The results in this paper present a prediction model for the first requirement check in the airbag design process, i.e. the volume simulation, and the framework to develop databases that can be used for similar requirement checks. With presented real-time prediction ability, the designers will be able to carry out the required number of design iterations among design tools or design teams in a faster and more effective way.

The findings in this paper also reveal the advantages and disadvantages of using automatic shape generation for deep learning databases. It can be asserted that engineering-based databases have special characteristics and therefore need special treatments in terms of machine learning algorithms used with them. The disadvantage is the risk of having skewed training data. For example, some little change in the dimension of the bag produces very little difference in the image's pixels but the output may be affected significantly.

There are advantages to engineering databases as well. For example, because all the images are created in an automated fashion, a lot of pre-processing that is common with databases can be saved. Taking screenshots by code in the CAD environment makes it possible to lay all the constant pixels in one (pixel) position which can save a lot of training time in the process. This is something that takes a lot of processing time in other databases. Not utilising this method requires extensive data augmentation that artificially expands the size of training databases by creating modified versions of the images such as rotated, distorted, and resized images, etc.

The contribution of the paper to the literature is discussed in the Introduction section. However, the implication of the industrial contributions can also be added here. As discussed earlier, during the design process the designers can benefit from a good enough estimation to have an idea about the effects of taken decisions. Often designers in the conceptual phase want to get a rough idea for a simple change (like a parameter on the geometry). This kind of need highlights the importance of proposed prediction tools that will accelerate those small-fix iterations. From the managerial point of view, having such predictors in the design process is also important. This is not only helpful in reducing the design lead time and freeing the resources but also is an enabler for design people to perform performance analysis (like CAE) and by doing so, prevent unnecessary iterations between design teams. Integrating such predictors in a CAD environment is trivial and therefore is not included in this paper. However, a graphical user interface increases the accessibility of using and maintaining such tools in the industry.

Although it can be argued that training and testing of such predictors take time, this is only a one-time investment. Once the model is trained the saved time compensates for the initial investment. In addition, many companies already store a lot of CAD information from past projects that can be utilised. Furthermore, by having a constant feed into the training database and adding the final version of every project, these models can be maintained. Lastly, the paper reaches the findings by the use of off-the-shelf tools and open-source codes which helps to improve the repeatability of this framework on similar problems in the industry.

Naturally, there are some limitations to the purposed framework as well. This limitation lies in the ability of 2D images in representing the design input space. From a usability perspective, some properties such as form, strain, or final position that are more representable by the geometrical variation can be used more effectively by this framework. Therefore, predicting objectives that are more affected by other aspects of product development such as material choices, energy, and information processing mechanisms in the product will face higher errors. The error margin on approximating models (such as the presented one here) will never be zero and thus these predictors will never be able to replace conventional analysis methods but rather be used as a complementary tool in the design toolbox.

Another limitation that also affects the generalisation of the proposed framework is that the utilised dynamic relaxation method in this paper cannot be applied to any simulation problem. More studies are necessary to investigate the application of dynamic relaxation in labelling other types of problems as well other objectives such as stress or fatigue. However, the framework can be generalised to many existing geometry-sensitive simulations. As mentioned in the previous sections, this method has been applied to specific form-finding problems of membranes and unstable structures such as hanging cables, chains, and domes.

Future work can involve applying the framework to more diverse simulation problems. Exploring the abilities and limitations of dynamic relaxation in labelling engineering databases. Moreover include more physical aspects of the geometry by considering cloud-based or voxel-based representation as an input data type for training AI models.

8. Conclusions

An iterative and simulation-driven design process is studied and the associated development lead time problem is addressed. To avoid parameterisation and high dimensionality, *real-time prediction model* as a hypothetical solution that includes the use of data-driven design together with a labelling process for the database is presented. Performed literature review of the latest data-driven methods in engineering design shows a gap in the literature for using real-case scenarios with complex geometry, a lack of engineering databases for benchmarking, and also methodologies for building such databases, i.e. labelling these databases. The framework presented in this paper uses dynamic relaxation in Grasshopper to estimate the volume as an objective for the curtain airbag design case. By reducing simulation time, this process made it possible to label a large number of CAD geometries with a small amount of finite element analysis. An image-based machine learning library with Convolutional neural networks gives a 90% accurate response in several milliseconds. Being able to analyse an idea fast when and where CAD work is being performed, significantly reduces iterations and the development lead time which shows the importance of such a prediction tool. This ability could also capture downstream implicit knowledge and make it available in the early phases. Additionally, the paper contributes to the literature by providing an engineering-based database as well as a framework on how to acquire it. This case study shows using accessible CAD screenshots has several advantages over imaging methods from products. The possibility of being integrated with usual CAD environments makes such design tools maintainable easily but new studies are necessary to outline the possibilities and limitations of this tool.

Acknowledgments

This work has been carried out within the project Butterfly Effect in the school of engineering, Jönköping University. The authors would like to acknowledge everyone in Jönköping University who was involved in this project in any way, especially Dr. Joel Johansson and Dr. Tim Heikkinen who made this work possible.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

The authors would like to acknowledge the staff in Autoliv® in Sweden for their participation in the project and also the Swedish Knowledge Foundation (*KK-Stiftelsen* with grant number 20180189) for the financial support.

References

Ahmed, Eman, and Mohamed Moustafa 2016. "House Price Estimation From Visual and Textual Features." arXiv preprint arXiv:1609.08399.

André, Samuel. 2017. "Supporting the Utilization of A Platform Approach in The Engineer-to-Order Supplier Industry." PhD diss., Jönköping University, School of Engineering.

Angulu, Raphael, Jules R. Tapamo, and Aderemi O. Adewumi. 2018. "Age Estimation Via Face Images: a Survey." *EURASIP Journal on Image and Video Processing* 2018 (1): 1–35. doi:10.1186/s13640-018-0278-6.

Arjomandi Rad, Mohammad. 2020. *Reducing Development Lead Time in Iterative, Simulation-driven Design Processes*. Jönköping: Jönköping University.

Arjomandi Rad, Mohammad. 2022. "Airbag Cad to Cae Volume." https://www.kaggle.com/mohammadrad/airbag-cad-to-cae-volume, Data files.

Belaid, Mohamed Karim, Maximilian Rabus, and Ralf Krestel. 2021. "Crashnet: An Encoder–decoder Architecture to Predict Crash Test Outcomes." *Data Mining and Knowledge Discovery* 35 (4): 1688–1709. doi:10.1007/s10618-021-00761-9.

Briard, Tristan, Camille Jean, Améziane Aoussat, and Philippe Véron. 2023. "Challenges for Data-driven Design in Early Physical Product Design: A Scientific and Industrial Perspective." *Computers in Industry* 145: 103814. doi:10.1016/j.compind.2022.103814.

Cao, Jiajun, Qingbiao Li, Liping Xu, Rui Yang, and Yuejin Dai. 2021. "Non-Parametric Surrogate Model Method Based on Machine Learning. *Preprints*, https://www.preprints.org/manuscript/202104.0762/v2.

Chang, Angel X., Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, et al. 2015. "Shapenet: An Information-Rich 3D Model Repository." arXiv preprint arXiv:1512.03012.

Chiarello, Filippo, Paola Belingheri, and Gualtiero Fantoni. 2021. "Data Science for Engineering Design: State of the Art and Future Directions." *Computers in Industry* 129: 103447. doi:10.1016/j.compind.2021.103447.

Crevier, Daniel. 1993. *AI: The Tumultuous History of the Search for Artificial Intelligence*. New York, NY: Basic Books, Inc.

Cunningham, James D., Timothy W. Simpson, and Conrad S. Tucker. 2019. "An Investigation of Surrogate Models for Efficient Performance-based Decoding of 3d Point Clouds." *Journal of Mechanical Design* 141 (12): 121401. doi:10.1115/1.4044597.

Deshpande, Saurabh, Jakub Lengiewicz, and Stéphane Bordas. 2021. "Fem-Based Real-Time Simulations of Large Deformations with Probabilistic Deep Learning." arXiv preprint arXiv:2111.01867.

Dong, C., Y. Yang, Q. Chen, and Z. Wu. 2022. "A Complex Network-based Response Method for Changes in Customer Requirements for Design Processes of Complex Mechanical Products." *Expert Systems with Applications* 199: 117124. doi:10.1016/j.eswa.2022.117124.

Du, Qiuwan, Tianyuan Liu, Like Yang, Liangliang Li, Di Zhang, and Yonghui Xie. 2022. "Airfoil Design and Surrogate Modeling for Performance Prediction Based on Deep Learning Method." *Physics of Fluids* 34 (1): 015111. doi:10.1063/5.0075784.

Ferreira, José Pedro d, and James Bell. 2020. "Deep Learning Image Analysis for Angular Measurements in Wind Tunnels." In *AIAA Scitech 2020 Forum*, Orlando, FL, 1979. doi:10.2514/6.2020-1979.

Fuhg, Jan N. 2019. "Adaptive Surrogate Models for Parametric Studies." arXiv:1905.05345 [cs, stat]. http://arxiv.org/abs/1905.05345.

Guo, Xiaoxiao, Wei Li, and Francesco Iorio. 2016. "Convolutional Neural Networks for Steady Flow Approximation." In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, CA, 481–490. doi:10.1145/2939672.2939738.

Heikkinen, Tim. 2021. "Extended Product Models Supporting Multidisciplinary Design Automation." PhD diss., Jönköping University, School of Engineering.

Hein, Phyo Htet, Elisabeth Kames, Cheng Chen, and Beshoy Morkos. 2022. "Reasoning Support for Predicting Requirement Change Volatility Using Complex Network Metrics." *Journal of Engineering Design* 33 (11): 811–837. doi:10.1080/09544828.2022.2154051.

Hirth, A., A. Haufe, and L. Olovsson. 2007. "Airbag Simulation with LS-DYNA Past-Present-Future." In *6th European LS-DYNA User Conference*, Gothenburg, Sweden, May 29th & 30th. Citeseer.

Jiao, Roger, Sesh Commuri, Jitesh Panchal, Jelena Milisavljevic-Syed, Janet K. Allen, Farrokh Mistree, and Dirk Schaefer. 2021. "Design Engineering in the Age of Industry 4.0." *Journal of Mechanical Design* 143 (7): 070801. doi:10.1115/1.4051041.

Johansson, Joel. 2011. "How to Build Flexible Design Automation Systems for Manufacturability Analysis of the Draw Bending of Aluminum Profiles." *Journal of Manufacturing Science and Engineering* 133 (6): 061027. doi:10.1115/1.4005355.

Johansson, Joel. 2014. "A Feature and Script Based Integration of Cad and Fea to Support Design of Variant Rich Products." *Computer-Aided Design and Applications* 11 (5): 552–559. doi:10.1080/16864360.2014.902687.

Khadilkar, Aditya, Jun Wang, and Rahul Rai. 2019. "Deep Learning-based Stress Prediction for Bottom-up Sla 3d Printing Process." *The International Journal of Advanced Manufacturing Technology* 102 (5): 2555–2569. doi:10.1007/s00170-019-03363-4.

Kim, Sangpil, Hyung-gun Chi, Xiao Hu, Qixing Huang, and Karthik Ramani. 2020. "A Large-Scale Annotated Mechanical Components Benchmark for Classification and Retrieval Tasks with Deep Neural Networks." In *European Conference on Computer Vision*, Glasgow, UK, 175–191. Springer. doi:10.1007/978-3-030-58523-5_11.

Koch, Sebastian, Albert Matveev, Zhongshi Jiang, Francis Williams, Alexey Artemov, Evgeny Burnaev, Marc Alexa, Denis Zorin, and Daniele Panozzo. 2019. "Abc: A Big Cad Model Dataset for Geometric Deep Learning." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Long Beach, CA, 9601–9611. doi:10.1109/CVPR.2019.00983.

Li, Yu, Hu Wang, and Juanjuan Liu. 2017. "Can CNN Construct Highly Accurate Models Efficiently for High-Dimensional Problems in Complex Product Designs?" arXiv preprint arXiv:1712.01639.

Maier, Jakob F., Claudia Eckert, and P. John Clarkson. 2019. "Experimental Investigation of the Implications of Model Granularity for Design Process Simulation." *Journal of Mechanical Design* 141 (7): 071101. doi:10.1115/1.4042571

Messner, Mark C. 2020. "Convolutional Neural Network Surrogate Models for the Mechanical Properties of Periodic Structures." *Journal of Mechanical Design* 142 (2): 024503. doi:10.1115/1.4045040.

Mohammad, Arjomandi Rad, Kent Salomonsson, Mirza Cenanovic, Henrik Balague, Dag Raudberget, and Roland Stolt. 2022. "Correlation-based Feature Extraction From Computer-aided Design, Case Study on Curtain Airbags Design." *Computers in Industry* 138: 103634. doi:10.1016/j.compind.2022.103634.

Nie, Zhenguo, Haoliang Jiang, and Levent Burak Kara. 2020. “Stress Field Prediction in Cantilevered Structures Using Convolutional Neural Networks.” *Journal of Computing and Information Science in Engineering* 20 (1): 011002. doi:10.1115/1.4044097.

Nunez, Marco, Varun Chander Datta, Arturo Molina-Cristobal, Marin Guenov, and Atif Riaz. 2012. “Enabling Exploration in the Conceptual Design and Optimisation of Complex Systems.” *Journal of Engineering Design* 23 (10-11): 852–875. doi:10.1080/09544828.2012.706800.

Panchal, Jitesh H., Mark Fuge, Ying Liu, Samy Missoum, and Conrad Tucker. 2019. “Special Issue: Machine Learning for Engineering Design.” *Journal of Mechanical Design* 141 (11): 110301. doi:10.1115/1.4044690.

Rombouts, Jef, Geert Lombaert, Lars De Laet, and Mattias Schevenels. 2018. “On the Equivalence of Dynamic Relaxation and the Newton-Raphson Method.” *International Journal for Numerical Methods in Engineering* 113 (9): 1531–1539. doi:10.1002/nme.5707.

Secco, Ney Rafael, and Bento Silva de Mattos. 2017. “Artificial Neural Networks to Predict Aerodynamic Coefficients of Transport Airplanes.” *Aircraft Engineering and Aerospace Technology* 89 (2): 211–230. doi:10.1108/AEAT-05-2014-0069.

Shan, Songqing, and G. Gary Wang. 2010. “Metamodeling for High Dimensional Simulation-based Design Problems.” *Journal of Mechanical Design* 132 (5): 051009. doi:10.1115/1.4001597.

Simpson, Timothy W., J. D. Poplinski, Patrick N. Koch, and Janet K. Allen. 2001. “Metamodels for Computer-based Engineering Design: Survey and Recommendations.” *Engineering with Computers* 17 (2): 129–150. doi:10.1007/PL00007198.

Soo Ko, Byung. 2022. “Imagenet Classification Leaderboard.” https://kobiso.github.io/Computer-Vision-Leaderboard/imagenet.html, Data files.

Toro, Javier Villena, Anton Wiberg, and Mehdi Tarkian. October 2022. “Application of Optimized Convolutional Neural Network to Fixture Layout in Automotive Parts. Preprint, In Review. https://www.researchsquare.com/article/rs-2154629/v1.

Umetani, Nobuyuki, and Bernd Bickel. 2018. “Learning Three-dimensional Flow for Interactive Aerodynamic Design.” *ACM Transactions on Graphics* 37 (4): 1–10. doi:10.1145/3197517.3201325.

Vurtur Badarinath, Poojitha, Maria Chierichetti, and Fatemeh Davoudi Kakhki. 2021. “A Machine Learning Approach As a Surrogate for a Finite Element Analysis: Status of Research and Application to One Dimensional Systems.” *Sensors* 21 (5): 1654. doi:10.3390/s21051654.

Wang, Jun, Sonjoy Das, Rahul Rai, and Chi Zhou. 2018. “Data-driven Simulation for Fast Prediction of Pull-up Process in Bottom-up Stereo-lithography.” *Computer-Aided Design* 99: 29–42. doi:10.1016/j.cad.2018.02.002.

Wang, G. Gary, and Songqing Shan. 2007. “Review of Metamodeling Techniques in Support of Engineering Design Optimization.” *Journal of Mechanical Design* 129 (4): 370–380. doi:10.1115/DETC2006-99412.

Wong, Jerry C. 2013. “Modeling of High Pressure Confined Inflatable Structures.” PhD diss.

Yoo, Soyoung, Sunghee Lee, Seongsin Kim, Kwang Hyeon Hwang, Jong Ho Park, and Namwoo Kang. 2021. “Integrating Deep Learning Into Cad/cae System: Generative Design and Evaluation of 3d Conceptual Wheel.” *Structural and Multidisciplinary Optimization* 64 (4): 2725–2747. doi:10.1007/s00158-021-02953-9.

Zhao, Xiaoyu, Zhiqiang Gong, Jun Zhang, Wen Yao, and Xiaoqian Chen. 2021. “A Surrogate Model with Data Augmentation and Deep Transfer Learning for Temperature Field Prediction of Heat Source Layout.” *Structural and Multidisciplinary Optimization* 64 (4): 2287–2306. doi:10.1007/s00158-021-02983-3.