energies

MDPI

Article

Modeling of TES Tanks by Means of CFD Simulation Using Neural Networks

Edgar F. Rojas Cala [1], Ramón Béjar [1], Carles Mateu [1], Emiliano Borri [1], Alessandro Romagnoli [2] and Luisa F. Cabeza [1,*]

[1] GREiA Research Group, Universitat de Lleida, Pere de Cabrera 3, 25001 Lleida, Spain; edgar.rojas@udl.cat (E.F.R.C.); ramon.bejar@udl.cat (R.B.); carles.mateu@udl.cat (C.M.); emiliano.borri@udl.cat (E.B.)

[2] School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore 639798, Singapore; a.romagnoli@ntu.edu.sg

* Correspondence: luisaf.cabeza@udl.cat

Abstract: Modeling of thermal energy storage (TES) tanks with computational fluid dynamics (CFD) tools exhibits limitations that hinder the time, scalability, and standardization of the procedure. In this study, an innovative technique is proposed to overcome the challenges in CFD modeling of TES tanks. This study assessed the feasibility of employing neural networks for TES tank modeling, evaluating the similarities in terms of structure and signal-to-noise ratio by comparing images generated by neural networks with those produced through CFD simulations. The results regarding the structural similarity index indicate that around 94% of the images obtained have a similarity index above 0.9. For the signal-to-noise ratio, the results indicate a mean value of 25 dB, which can be considered acceptable, although indicating room for improvement. Additional results show that our neural network model obtains the best performance when working with initial states close to the stable phase of the TES tank. The results obtained in this study are promising, laying the groundwork for a future pathway that could potentially replace the current methods used for TES tank modeling.

Keywords: CFD; neural network; machine learning; TES tank; simulation; modeling

check for updates

Academic Editor: Jiaqiang E

Received: 21 December 2024
Revised: 17 January 2025
Accepted: 20 January 2025
Published: 23 January 2025

Citation: Rojas Cala, E.F.; Béjar, R.; Mateu, C.; Borri, E.; Romagnoli, A.; Cabeza, L.F. Modeling of TES Tanks by Means of CFD Simulation Using Neural Networks. *Energies* **2025**, *18*, 511. https://doi.org/10.3390/en18030511

Copyright: © 2025 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

Thermal energy storage (TES) systems are components that can store thermal energy at a physical location for periods of time to be released when it is needed. With the increase in energy demand during the last decade and the need for more sustainable energy production, TES has found large potential for implementation in a wide variety of applications, whether for solar installations, waste heat recovery, or heating/cooling systems for both the residential and non-residential sectors [1–6].

This versatility has allowed for a huge variety of TES system designs created by different researchers or industries. However, creating a large variety of designs does not help in the standardization and scalability of TES technologies.

The need to develop a specific design for a targeted application can increase the costs of TES solutions, requiring specialized tools as well as qualified personnel to carry out the design process [1,7–10]. The TES system design process can be approached from several points of view, one of which is modeling and simulation, which requires a separate area of expertise as well as certain resources [11–15].

The use of artificial intelligence to model and simulate physical systems is now widespread, and satisfactory results have been obtained in interpreting the behaviors of a wide variety of these physical systems [15–17]. In this sense, the literature shows that there is high interest in the use of artificial intelligence to model the behaviors of fluids. In their study, Chickerur et al. [18] implemented convolutional neural networks (CNNs) to estimate the velocity field of different objects in a 2D representation. Also in their study, Usman et al. [19] proposed the use of a Fourier neural operator (FNO) to implement a neural network capable of modeling the pressure behavior in a system composed of a circular body and an incompressible fluid.

In the case of TES systems with artificial intelligence, the literature shows extensive use of CFD simulations with artificial intelligence as a hybrid mode to predict the behaviors of TES systems. In the study by Madhankumar et al. [20], various classical machine learning methods were proposed, including multiple linear regression (MLR), random forest (RF), extreme gradient boosting (XGBoost), and artificial neural networks (ANNs), to predict the outlet temperature and efficiency of a corrugated absorber plate solar collector. The study highlights the high level of predictive accuracy of the hybrid model with ANN compared to the other methods studied. Similarly, the findings of Yuanji et al. [21] demonstrated the fidelity of a rapid prediction model based on a simple multilayer percetron (MLP) architecture for an ANN to obtain the fusion friction and heat storage in a latent heat TES tank. The study reported a deviation of less than 10% between the proposed model and data simulated using CFD. An analysis by Burak et al. [22] showed the use of a hybrid method that combined CFD simulations with artificial intelligence methods, namely the machine learning method XGBoost, to validate hypotheses regarding the orientation of a container for PCM melting. The study results indicated a significant advantage when using these methods to validate outcomes. Dhakane et al. [23] proposed using an ANN trained with CFD data to obtain the same behavior predictions for bubble columns, and Nadda et al. [24] used a similar approach to predict the hydrodynamics of fluidized beds. A totally different approach to using artificial intelligence for CFD is that of Rubio et al. [25], who used an ANN to create an expert system to help predict the behaviour of the CFD solver.

Considering the literature, this study proposes a novel alternative for modeling TES tanks, which is currently often carried out using CFD tools. In this regard, the use of more complex artificial neural networks (ANNs) to create digital twins [26] was considered, with a higher representative power that the MLP networks used in previous studies, with the aim of generating fluid dynamics models in a manner like computational fluid dynamics (CFD) software. These digital twins will be created by training artificial neural networks with simulation data, allowing these networks to learn and replicate the dynamic behaviors of fluids in TES tanks.

The development of the above proposal can generate a considerable reduction in terms of time when carrying out TES tank simulations, and increase scalability and standardization, which would allow the design of TES tanks to be performed in a faster way. This time reduction is key to using these models for optimizing TES tank design, as any optimization algorithm (genetic algorithms, particle swarm optimization, etc.) will require hundreds or thousands of TES tank simulations.

To accomplish this, an artificial neural network (ANN) was trained in a minimalist latent heat storage tank design using COMSOL simulation software v5.3a [27]. The neural network was built and trained using the Python programming language, together with libraries designed for the development of neural networks, such as Fastai [28,29] and Pytorch [30]. Finally, a comparison was carried out betwee n the images generated by the neural network and the ones generated with COMSOL software to determine the feasibility and prediction accuracy of the implemented neural network model.

2. Methodology

The goal of this study is to use an ANN to simulate the temperature change in a small TES tank, modeled using CFDs. As neural networks, like all supervised learning techniques, require being trained with labeled data, first we built a dataset containing pairs of temperature situations of the TES tank. In this study, we focused on the discharging phase. These temperature states were represented as images depicting the heatmap, with each member of the pair representing, respectively, the state at time 0 and the state after a defined time step t; in our case, we chose a 3 s time step. Then, the neural network architecture, based on an autoencoder architecture, was implemented and trained to model the temperature state change experienced by the TES tank during that time step. For the sake of simplicity and to have reasonable computation times for the neural network training and dataset generation, we chose a small 2D TES tank storing thermal energy as the TES tank to be modeled.

Besides temperature maps, other features must be considered, namely, the initial temperatures for the heat transfer fluid and for solid storage and the Reynolds number. These additional features will also be part of the dataset.

In summary, the final structure of the dataset was as follows: for each time step, we had an image depicting the initial temperature heatmap of the TES tank, three additional features, and another image depicting the temperature heatmap of the TES tank after the time step has passed.

2.1. Dataset Generation

To obtain the necessary dataset, we resorted to using CFD simulations to generate the heatmaps. CFD is a commonly used method to model this kind of problem. The CFD software we used is COMSOL [27].

We used CFD simulation software, generating simulations for different values of the initial parameters and generating images of the same tank every second for 320 s. To create the CFD simulations in COMSOL, we started by designing the structure of the 2D TES tank. In order to reduce the design variabilities, the TES tank chosen was a sensible heat storage tank in the discharge process (i.e., with materials without phase change and with heat losses) [1]. The tank was composed of an outer shell (not considered in the simulations) of rectangular shape, a liquid that flows through the tank to transfer the energy stored in the TES tank to the outside, and a circumference in the center of the tank, composed of a solid material in which an initial thermal energy is stored. A schematic of the above description of the TES tank with the used dimensions, determined by variables "a", "b", "c", "D", "H", and "L", can be seen in Figure 1. The value for each variable mentioned above is presented in Table 1.

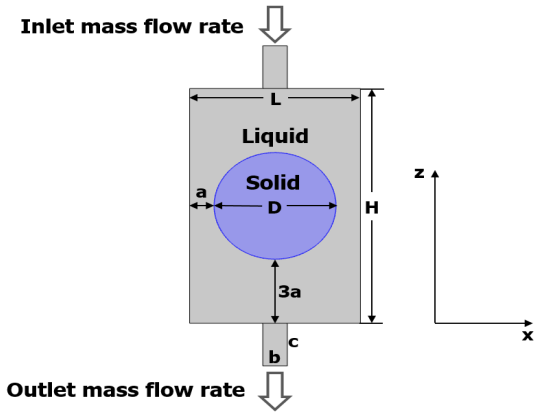


Figure 1. Design proposal for a two-dimensional TES tank.

Table 1. Values used for the different variables that define the specific TES tank design (see Figure 1) used to train the neural network for this study.

Value	Variable
0.02 m	a
0.02 m	b
0.04 m	c
0.01 m	D
0.14 m	L
0.22 m	H

The model presented above was loaded into COMSOL software which, as mentioned above, also requires a mesh of nodes to understand the structure of the proposed design. The shape of the mesh is crucial, as it determines how accurate the simulation result is at all points in space. In this case, it is important to emphasize the shape of the mesh in the center of the solid circle, as the resulting images may not reliably model the temperature behavior in these regions due to the spacing between nodes.

Having defined how COMSOL software would interpret the two-dimensional space, it was necessary to make some assumptions. These assumptions not only help to simplify the model but also allow COMSOL software to model the behavior over time inside the tank. These assumptions are as follows:

- The tank has no thermal energy losses.
- No friction on the outer walls of the tank.
- The fluid has a laminar regime in 2D.
- The fluid is not turbulent.
- The fluid has a Newtonian behavior.
- The fluid is isotropic and incompressible.
- Computationally, the fluid is a continuous medium.
- The edge of the circle representing the solid is frictionless.
- The solid is an isotropic material.

Figure 2 shows the mesh used in COMSOL for the proposed design. As explained above, for the CFD simulation software, COMSOL was used to generate the initial design to train the network.

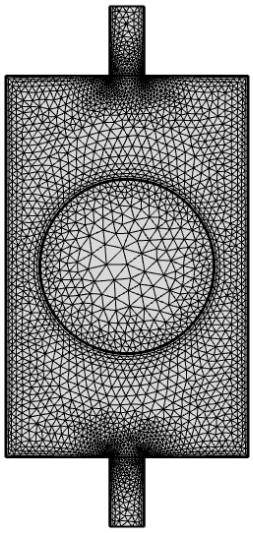


Figure 2. View of the mesh used to design a TES tank that was simulated using the COMSOL tool. This COMSOL model was used to generate the data for training the neural network.

COMSOL software uses a series of equations to solve the behavior at each node of the grid. Equation (1) [31,32] establishes the energy behavior for the fluid flow inside the tank; this uses the coordinate plane "z", "x", as presented in Figure 1. Equation (2) [31,32], $\frac{k}{\rho C_p}\left(\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial T}{\partial r}\right)+\frac{1}{r^2 \sin\theta}\frac{\partial}{\partial \theta}\left(\sin\theta\frac{\partial T}{\partial \theta}\right)\right)=\frac{\partial T}{\partial t}$, represents the heat transfer inside the solid circle. The equation uses polar coordinates "r" and "θ", which indicate each point in space inside the circle for which the behavior is to be known; k, ρ, μ, and C_p are properties of the fluid and solid that will be explained later.

$$\rho C_p\left(\frac{\partial T}{\partial t}+u\frac{\partial T}{\partial x}+u\frac{\partial T}{\partial y}\right)=k\left(\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial y^2}\right) \quad (1)$$

$$\frac{k}{\rho C_p}\left(\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial T}{\partial r}\right)+\frac{1}{r^2 sin\theta}\frac{\partial}{\partial \theta}\left(sin\theta\frac{\partial T}{\partial \theta}\right)\right)=\frac{\partial T}{\partial t} \quad (2)$$

Other elements to be considered were the properties of the materials of the TES tank. As mentioned earlier, this study used only three variable parameters to train the neural network. However, COMSOL software requires the definition of multiple parameters to generate an optimal simulation of the process and uses these properties to solve the equations governing tank behavior.

Table 2 presents the values used in the COMSOL simulations. These values were not included in the initial neural network training in this study but will be incorporated in future work.

Table 2. Physical design parameters used in the COMSOL simulation tool for the design of the TES tank. See Figure 1.

Property	Variable (or Name)	Value
Fluid density	$\rho(rho)$	998.2 kg/m^3
Fluid dynamic viscosity	$\mu(mu)$	0.001002 Pa·S
Fluid thermal conductivity	K	0.5861 W/m·K
Fluid specific heat capacity	c_p	4183 J/kg·K
Solid density	$\rho_{co}(rho)$	2300 kg/m^3
Solid thermal conductivity	K_{co}	1.8 W/m·K
Solid specific heat capacity	c_{pco}	880 J/kg·K

The parameters that varied in value, both for training the neural network and for obtaining simulations in COMSOL software, were the initial temperature of the liquid, the initial temperature of the solid, and the fluid viscosity through the tank (i.e., Reynolds number). Before defining the range of values over which these properties of the tank materials would vary, a consideration was made. As we were modeling the discharging state, for the discharge to take effect, the initial liquid temperature must necessarily be lower than the initial solid temperature; so, only valid combinations satisfying these constraints were used. Of this range of possible values, we took five possible values for each feature, covering the full range, with a total of 125 possible combinations. The chosen values are shown in Table 3.

Table 3. Variable design parameters that were used for CFD modeling in COMSOL simulations and the neural network.

Input Property	Variable Name	Values Considered
Reynolds number	Re	[200, 350, 500, 700, 950]
Initial temperature of solid	Tc	[333, 343, 353, 363, 373] K
Initial temperature of liquid	Tw	[283, 293, 303, 313, 323] K

Finally, for each one of these 125 combinations of values, a CFD simulation was carried out in COMSOL, with each lasting 320 s (more than enough time for the natural response of the TES tank once unloaded). The simulation generated one heatmap snapshot per second, representing the behavior of the temperature inside the tank in that second of simulated time. Those images were then used to create the dataset. Each sequence of images with the same parameters (R, Tw, Tc) was split into pairs of images, each pair consisting of images taken 3 s apart. These pairs constituted a dataset containing an initial state (image at initial time) and a resulting state (image at time plus 3 s). Figure 3 shows this procedure graphically.

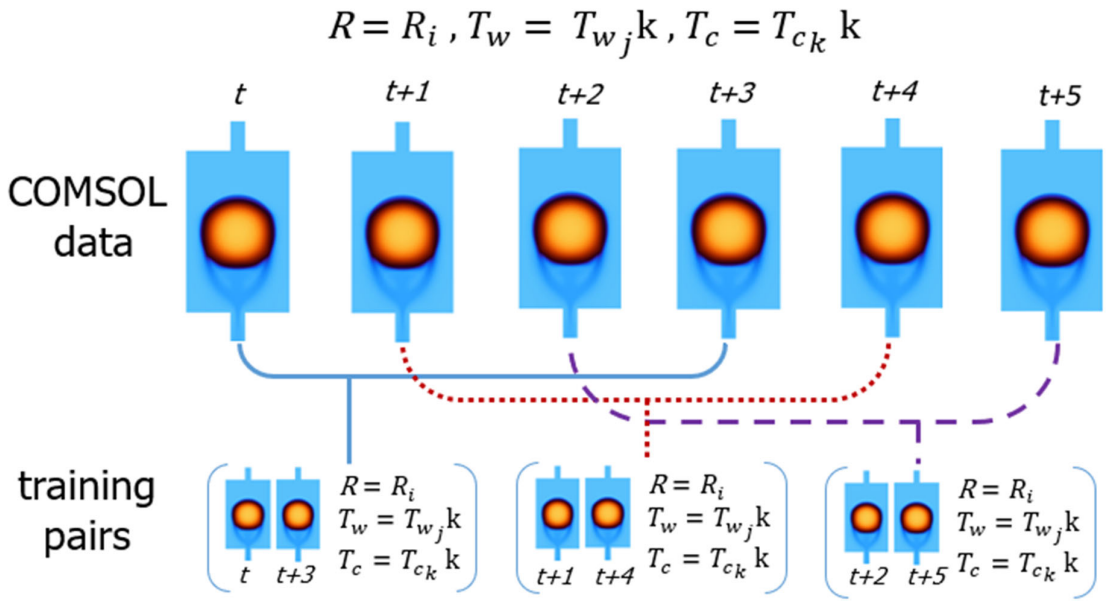


Figure 3. Construction of the training dataset from the COMSOL CFD data. The dataset used is comprised of pairs of images of the state of the TES tank separated by δ time, in this case δ = 3 s, with the pertinent parameters: Reynolds number (R), initial liquid temperature (T_w), and initial solid temperature (T_c). From COMSOL, we obtained long sequences of 320 images that can be paired, giving 317 pairs.

2.2. Neural Network Design

To describe the neural network architecture in detail, we consider first its inputs and outputs, and later move on to explain the inner architecture and layers. Our neural network receives, as its input, an image representing the temperature state of the tank being modeled at a given point in time. This image is at a 200 × 320 pixel resolution. Also, an additional set of scalar parameters is received, namely, the heat transfer liquid temperature, the solid temperature, and the Reynolds number. As the output, the neural network produces an image, of the same size as the input image, which is a prediction of the temperature map of the tank 3 s later. A diagram of this can be seen in Figure 4.

Now that we have established the neural network input and output parameters, we can focus on the inner architecture details. The neural network architecture is loosely based on a 2D convolutional autoencoder [33]. Roughly speaking, it can be described as a structure akin to a double funnel with the narrow sides pointing to each other, with each funnel being a 2D convolutional neural network (CNN). The first funnel (CNN [34]) receives highly dimensional data, the temperature map image, and gradually reduces its dimensionality, extracting the really important features that define that input data. The second funnel (DN [35]) receives, as its input, these compressed data and expands them, so it ends with the same dimensionality as it had in the input. The two funnels are named the encoder (the input funnel) and the decoder (the output funnel). Our architecture

departs from traditional autoencoder architectures as we added some additional inputs for additional features, namely, the fluid liquid temperature, solid temperature, and Reynolds number. These inputs skip the encoder and are passed on directly to the decoder. Figure 5 gives a more detailed view of the neural network architecture used in this study. The detailed layer architecture is explained as follows:

- **Encoder (also called contraction)**: This stage consists of a series of convolutional layers that reduce the dimensionality of the input data while interpreting patterns in that data; the reduction is what allows the network to generalize patterns. It is composed of Conv2D [36] layers with ReLu [37,38] activation functions and MaxPool layers [39].
- **Parameter layer:** This layer has three input parameters corresponding to the three variables of the different simulations: the Reynolds number, initial temperature of the fluid, and initial temperature of the solid. This layer is responsible for obtaining the relationship parameters between these three variables. The parameter layer is then concatenated with the encoder layer to model the relationship patterns between the image and the variables.
- **Decoder (extension):** This is the last stage of the process and is responsible for reconstructing the dimensionality of the image, considering the learned patterns and the output images on which it is trained. The result of this stage is data with the same dimensions as the input data, in this case, an image of 200 × 320 pixels.

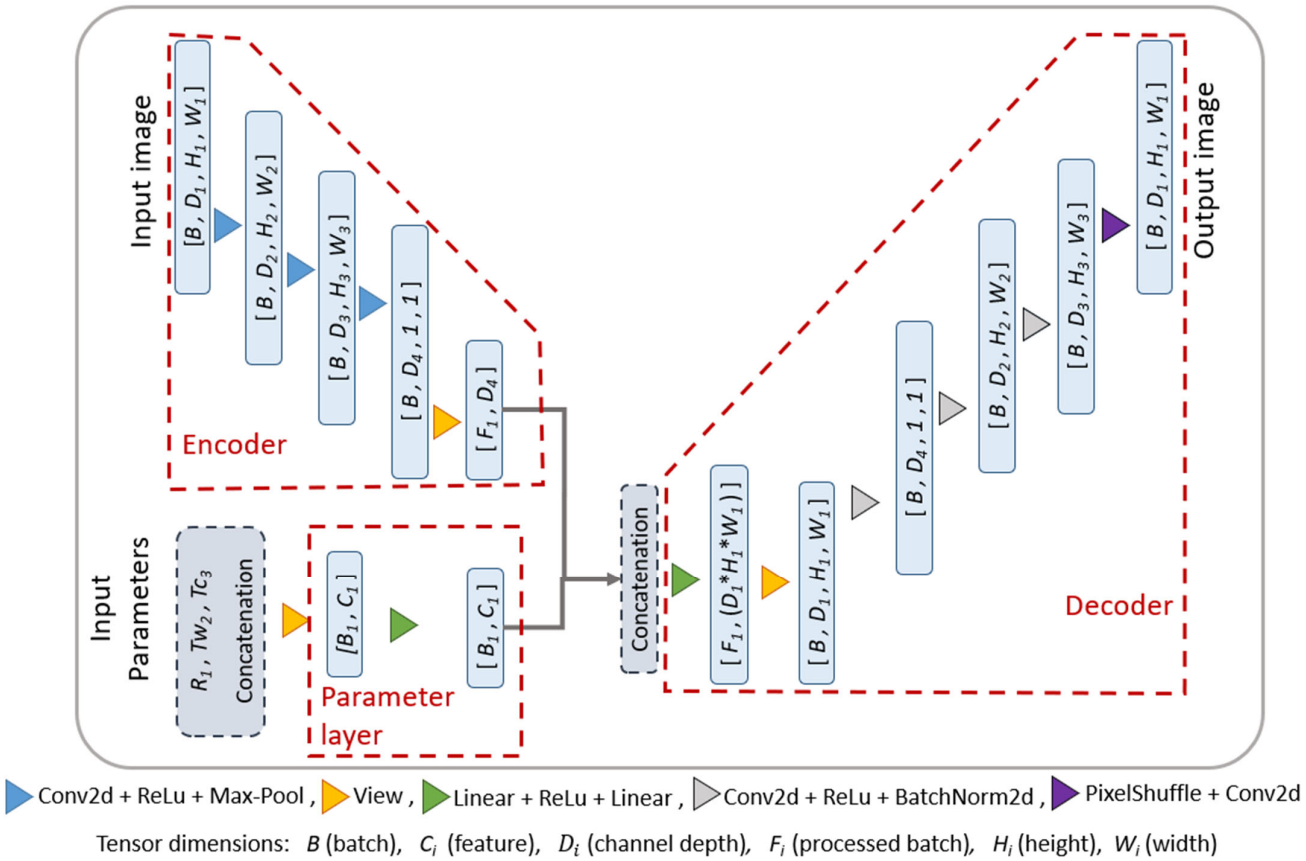


Figure 4. Proposed neural network architecture for NeNeCFD. It consists of an encoder (**top left**), a decoder (**right**), and a parameter layer (**bottom left**). Each neural layer shows the dimensions of the tensors in the layer output (B: batch, C_i: feature, D_i: channel depth, F_i: processed batch, H_i: height, W_i: width).

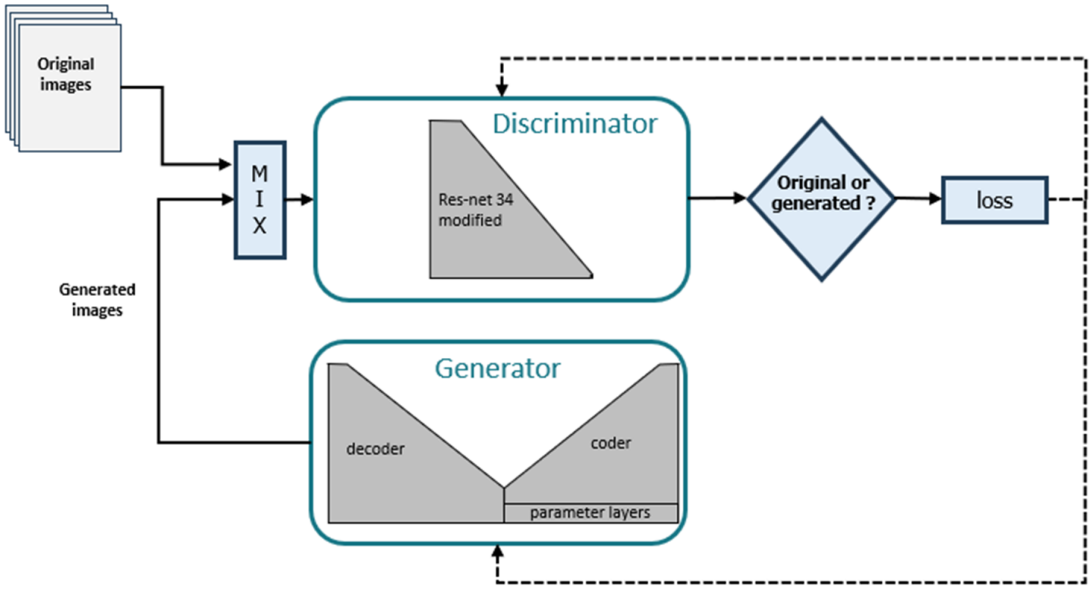


Figure 5. GAN (generative adversarial network) training method for neural networks. The output of the generator is used to train a discriminator network (a binary classifier) together with the original CFD images. Once the discriminator is good at distinguishing the original or created images, it is used as a loss function to retrain the generator network. This training lasts until the discriminator loses accuracy as a loss function. We then switch back to train the discriminator again. We continue looping the training of both networks until we reach a threshold.

2.3. Adversarial Neural Network Training

Besides the architecture of the neural network used in this work, another important aspect to consider is the training method used. In our case, we resorted to using generative adversarial network (GAN) training. This training method is based on the idea of co-training not one but two neural networks, where each one of them is trained to try to beat the other one. Regular one network training did not work in our case, as a key component of neural network training, the existence of a loss function, was not easily achievable. The loss function should have measured the difference between the generated temperature map image and a predefined real one, the ground truth heatmap image. Most existing functions measure the average color differences between images, but sometimes equally distributed errors of opposite signs cancel each other, thus making the loss measure poorly accurate. By training a neural network to discriminate and measure differences between images, we built a robust problem-oriented loss function.

In our case, we have, on the one side, our model neural network, which we trained to be able to predict the final state of the TES tank 3 s after a given initial state, and on the other side, we trained another neural network, a classifier, which tried to discern if a pair of heatmap images (initial state and 3 s later state) were generated by our neural network or if they belonged to the initial training dataset (generated by CFD), i.e., it tried to discriminate between ground truth image pairs (CFD) and simulated image pairs. By using the second network, the discriminator, as a loss function when training the first neural network (our model), in the training of our model is focused on trying to "fool" the discrimination process by better mimicking the CFD software. Conversely, training the second neural network focuses on trying to better discriminate between our model results and the original CFD results. In the GAN training method, the auxiliary neural network is known as the "discriminator network" and the main neural network is known as the "generative network" [40–43]. In summary, the goal was to train our neural network so that it was able to generate responses that were so close to reality, which the discriminator was unable to distinguish between our responses and the training dataset ground truth.

In Figure 6, we show the GAN training. It shows the first step of the process, where first the discriminator network is trained using "real" data (data not generated by the main neural network, i.e., CFD data) and "model" data (data generated by the generator neural network). In this way, the discriminator network improves its ability to discriminate between each type of data: real (CFD) or model (generated).

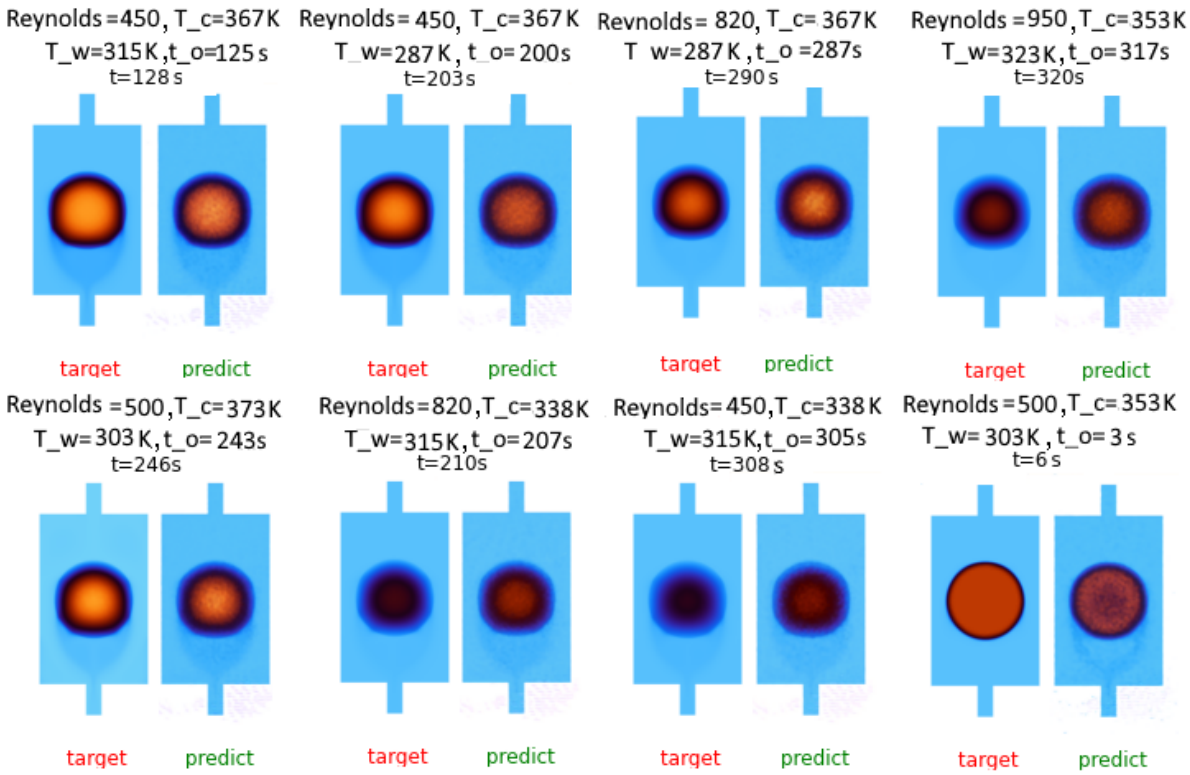


Figure 6. Part of the set of images generated by the neural network ("predict"), labeled by their initial values, accompanied by the images generated by COMSOL software ("target").

Using the previously trained discriminator network as a loss function, the generator network was trained, thus improving the generated images with the goal of fooling the discriminator network. This improvement in generation caused the loss function to become progressively lower, as is usual with neural network training. Once the loss was below a given threshold (our generator was better at fooling the discriminator), it was the moment to retrain the discriminator network again. This process of training and retraining of both networks was carried out several times, with the main objective that the generative network produced data so similar to the real CFD that the discriminator network lost the ability to discriminate between both sources.

To complete the details of the architecture used for the GAN training, we must also describe the architecture of the discriminator network. This network is based on an image classification model known as Resnet34 [44]. This model also uses convolutions to find patterns in the images; the difference is that the output of this model is classification values, in this case, a binary classification: "model" or "real". The discriminator neural network was trained beforehand to score images of the tank under investigation and to determine which images were generated by COMSOL software and which were generated by the generator network.

3. Results

After training the neural network, it could generate images that represented the state of a TES tank three seconds later (in terms of temperature), in the same way as a

CFD simulation. To evaluate the capability of the trained neural network, a new dataset was generated. An inference process was conducted on each newly generated data point, followed by a comparison between the images produced by CFD simulation and the images obtained from the neural network using computational image comparison tools.

Starting with the acquisition of new data to evaluate the trained neural network, the same TES tank design as described in Figures 1 and 3 was used. The initial values for liquid temperature, solid temperature, and Reynolds number were changed. These new values are given in Table 4. The CFD simulation software COMSOL was used to generate representative images of the temperature behavior in the tank every second over an interval of 320 s. In total, 5072 images were obtained, which formed the dataset used to test the effectiveness of the trained neural network.

Table 4. Values of TES tank characteristics that were used for the inference process in the neural network.

Property	Nomenclature	Initial Value
Reynolds number	Re	[450, 500, 820, 950]
Initial temperature of the solid	Tc	[338, 353, 367, 373] K
Initial temperature of the liquid	Tw	[287, 303, 315, 323] K

With the test dataset, an inference process was carried out on the neural network. The trained neural network was used to generate predictions of the behavior of the selected TES tank. The output data represent the state of the tank three seconds after the input state. Each generated image was then paired with its corresponding image produced by COMSOL software. Figure 6 shows some of these paired images. In each group, the image noted as "predict" in green at the bottom is the prediction generated by the neural network, while the image noted as "target" in red at the bottom is for the same time instant and input parameters as the predicted image, but in this case, generated by COMSOL software. Each set of paired images is annotated with its initial values and the simulation time it represents. While only a fraction of the 5072 images is displayed in Figure 7, a representative sample has been carefully chosen to provide the reader with insight into the neural network's performance.

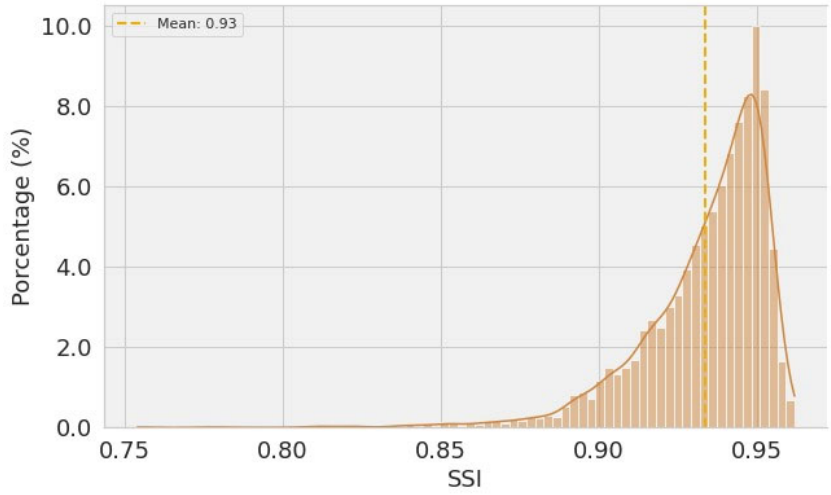


Figure 7. Histogram of the structural similarity index (SSI) percentage data with respect to the total number of images used in the neural network inference process.

In general, a substantial portion of the images produced by the neural network closely aligned with the expected outcomes, akin to the images generated by COMSOL in this case. Nevertheless, additional methods were essential to evaluate the network's efficacy

in generating images that accurately depict the TES tank behavior three seconds after an initial state.

The subsequent sections show the methodologies employed to assess this efficiency, considering the images generated throughout the neural network inference process. An analysis was carried out to analyze the differences between the images generated by the neural network and COMSOL software, using two different calculation methods, known as the structural similarity index and the peak signal-to-noise level [45–47].

The structural similarity index (SSI) is widely used for computational comparisons of distorted images. The mathematical formula defining this analysis method is shown in Equation (3), where μ_x is the mean of the real image (in this study, referring to the image generated by COMSOL software) and μ_y is the mean of the degraded image to be compared (in this study, referring to the image generated by the neural network). σ_x and σ_y are the standard deviations of the real image and the degraded image, respectively; σ_{xy} is the covariance between the two images; and C_1 and C_2 are constants. The structural similarity index (SSI) correlates with the analysis of three characteristics used by humans to comprehend an image: luminance, contrast, and structure. Luminance is related to the mean intensity of the images, μ_x and μ_y, while both contrast and structure are related to the covariance, σ_{xy}, and the standard deviations of the images, σ_x and σ_y. Finaly the SSI method is more precise than others, such as the mean square error, and can take values between -1 and 1, where values close to positive 1 indicate that the images tend to be equal [46,47].

$$SSI = \frac{(2\mu_x\mu_y + C_1) \times (2\sigma_{xy} + C_2)}{\left(\mu_x^2 + \mu_y^2 + C_1\right) \times \left(\sigma_x^2 + \sigma_y^2 + C_2\right)} \tag{3}$$

The other method used, peak signal-to-noise ratio (PSNR), is a variation of the mean squared error, which aims to measure the quality of a distorted image in relation to the original. It is expressed in decibels and its interpretation depends on a high decibel value; the higher the value, the more similar the distorted image is to the original. The PSNR is calculated using Equation (4), where 'b' is the number of bits per pixel of the image; M and N are the dimensions of the images in pixels; and 'x' and 'y' are the two images to be compared, the original and the distorted.

$$PSNR = 10log_{10}\frac{\left(2^b - 1\right)^2}{\frac{1}{M\times N}\sum_{j=1}^{M\times N}(x_j - y_j)^2} \tag{4}$$

Figure 7 shows a histogram representing the measurement of the structural similarity index (SSI) for the 5072 pairs of images obtained using COMSOL software and their corresponding digital twins obtained through the inference of the proposed neural network. The histogram illustrates that 93.67% of the images have a similarity of 90% or more to their COMSOL-generated counterparts. In addition, 50% of the total images achieve a similarity greater than 93.92%. Overall, the average similarity per image is 93.37%. Notably, no image falls below a similarity of 75.41% and the maximum similarity achieved by the neural network is 96.20%. The above data highlight the significant success in the use of neural networks for TES tank modulation.

On the contrary, when examining the analysis conducted using the PSNR method, a standard deviation of 2.89 dB is observed relative to the mean of 25.09 dB. This indicates that the neural network encounters difficulties in inferring certain types of data compared to others. This discrepancy may be attributed to two factors: the input parameters (Reynolds number, liquid temperature, solid temperature) and the influence of different time values used to construct the dataset. This aspect will be further discussed later in this article.

The graph depicted in Figure 8 underscores a notable observation regarding the images generated by the neural network in comparison to their COMSOL-generated counterparts. It becomes apparent that, as a collective trend, the images produced by the neural network exhibit a discernible level of noise. According to Equation (4), the calculation of PSNR adopts a logarithmic form, implying that its maximum value can be infinite. Values closer to 0 indicate a higher amount of noise between the compared pixels. In their studies, Martini et al. [46] and Wang et al. [47] mentioned that for the PSNR measurement method, values above 29 dB are considered acceptable, depending on the comparison context. In this study, many of the compared images exhibit PSNR values between 29 dB and 22 dB. Since each pixel represents the temperature at a point in the tank, noise levels within these ranges can affect the accuracy of the neural network presented in this analysis.

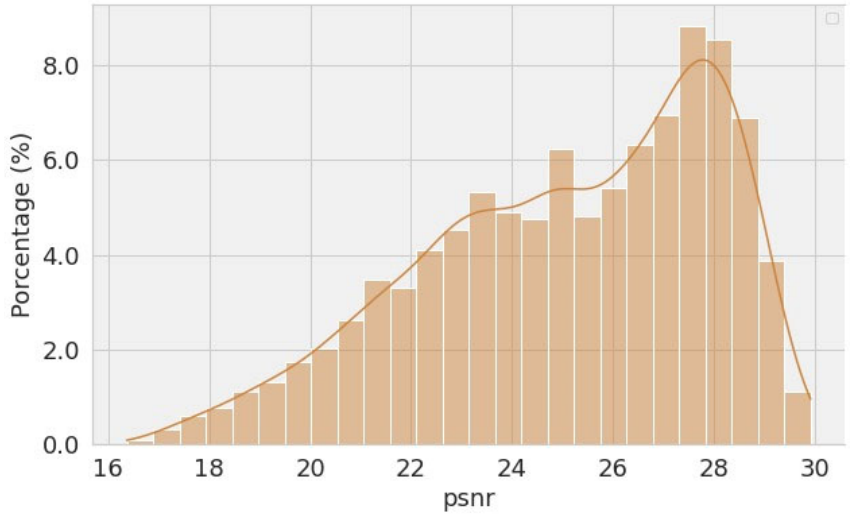


Figure 8. Histogram of the percentage comparison of the data obtained by the neural network and its homonyms obtained by COMSOL software with respect to the peak signal-to-noise ratio (PSNR).

In a general context, the graphs presented in Figures 7 and 8 give results that, although open to improvement, offer a very encouraging outlook, confirming the effectiveness of the implementation of neural networks in the modeling of TES tanks. Another type of analysis was conducted on the set of paired images to observe the effects of different input parameter values on the prediction efficiency of the neural network presented in this study. This analysis focused on the three additional input parameters used in this study: initial fluid temperature, initial solid temperature, and Reynolds number.

Figure 9 illustrates an analysis of the structural similarity index (SSI) and peak signal-to-noise ratio (PSNR) measures for the different values assumed by the three features used in this study. These values are normalized for better understanding. Regardless of the measurement method used to assess the correlation between the images produced by the neural network and those produced by COMSOL, Figure 10 clearly shows the influence of the 'solid temperature' characteristic (Tc_Norm) on the accuracy of the neural network. This influence extends to both structure fidelity and noise (pixels that deviate from the expected color). The results presented here contribute significantly to the ongoing discourse on the nuanced interplay between the input parameters and neural network performance in the field of TES tank modeling.

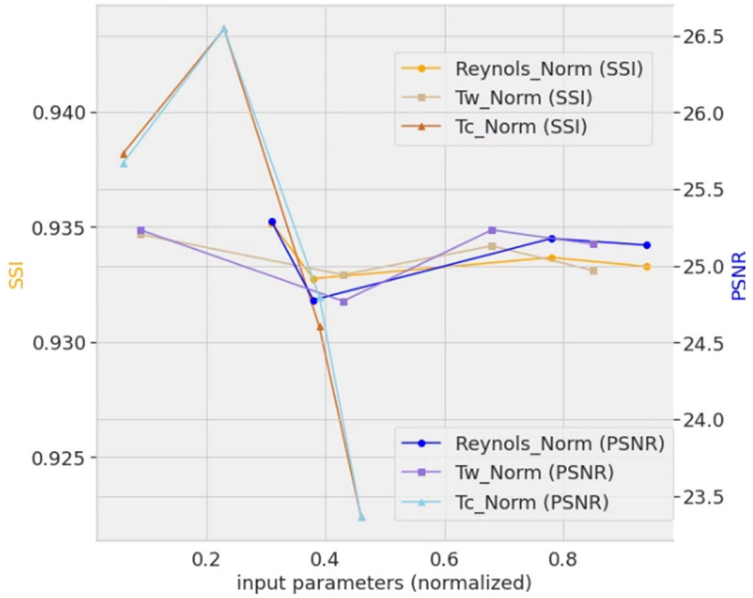


Figure 9. SSI (**left**) and PSNR (**right**) measurements with respect to the different normalized values of the input parameters taken in this study: Reynolds number (Reynols_Norm), initial temperature of the liquid (Tw_Norm), and initial temperature of the solid (Tc_Norm).

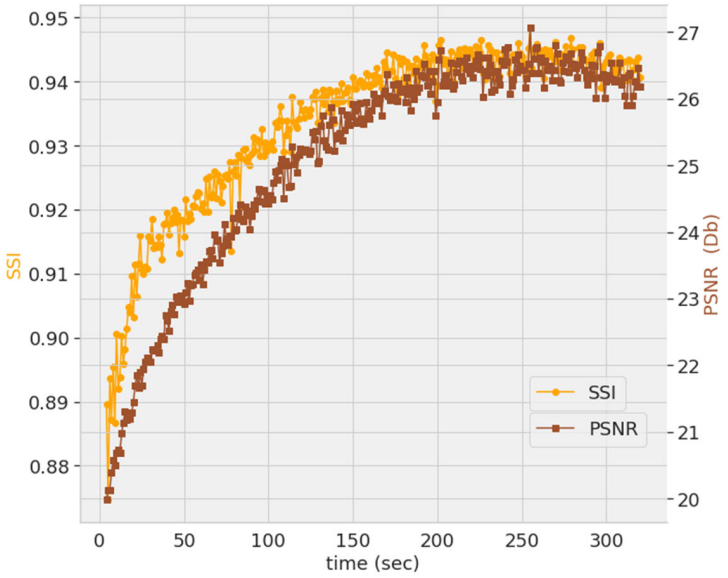


Figure 10. Measurement of the efficiency of the neural network by means of the SSI (**left**) and PSNR (**right**) methods with respect to the simulation time used to generate the dataset used in the training of the neural network.

Concerning the characteristics 'Reynolds number' and 'liquid temperature', Figure 9 reveals that these features exert a relatively modest impact on the neural network's efficiency. The behavior can be attributed, in part, to the observations depicted in Figure 7, where the region undergoing the most pronounced changes within each generated image of the dataset corresponds to the solid component (inner circle). This region exhibits diverse colors, particularly when the initial temperature of the solid is significantly higher than that of the liquid. Such effects may arise from dual factors. Furthermore, it is posited that the observed behavior in the solid temperature could potentially extend to the liquid temperature if the neural network were trained for a 'charging' process of the TES tank. In such a scenario, the region corresponding to the liquid is anticipated to display more color variations across the training image set.

While it is possible to attribute the decrease in accuracy efficiency, as previously discussed and evident in Figure 9, it is equally crucial to scrutinize the neural network's performance concerning the 320 s of simulation employed to generate the image dataset used in the training process. This consideration arises despite the fact that the neural network does not explicitly account for time. A cursory visual analysis of the dataset generated by the neural network, as depicted in Figure 6, reveals notable discrepancies in data generated at simulation times close to 0 s in comparison to their COMSOL-generated counterparts. The effects of the natural and forced responses of the TES tank discharge system employed in this study were systematically examined.

Figure 10 encapsulates the outcomes derived from the preceding analysis, where the aim was to observe the effects of different inferred initial states, in other words, the effects of predicting the behavior of the proposed TES tank for each second of the 320 s considered in the creation of the test dataset. In contrast to Figure 9, where the minimum SSI measure denotes an accuracy of 92.3%, Figure 10 discloses instances wherein the accuracy dips below 88% for select time values. Of noteworthy consideration is the observed progression wherein accuracy ascends and image noise diminishes with increasing time. However, a discernible deviation from this trend becomes apparent around the 250 s mark. The initial phase of this observed behavior is attributed to the system's forced response. In this context, the limited training cases fail to adequately guide the network in accurate interpretation, primarily due to the dataset construction. Specifically, a mere 125 data points represent this time value, coupled with 125 images. Given the fluctuations in the initial temperature of the concrete, the network grapples with the challenge of assimilating reliable patterns for generating images with a diminished margin of error.

For longer durations, where theoretically the TES tank system is in steady state, the analyses reveal an enhancement in the neural network's capacity to model akin to the COMSOL software. This improvement is attributed to the simulation phase where, in many cases, the images exhibit minimal variations, rendering it easier for the neural network to discern patterns of behavior. Consequently, this facilitates a more optimal training experience for images with similar structures and compositions.

In the exploration of how time impacts the efficiency of the neural network, a subsequent analysis was conducted to scrutinize the influence of different time values on the network across the diverse ranges inherent in input parameters of the TES tank. Figure 11 elucidates the noise engendered in the comparison of each image over time, segmented for each variable—Reynolds number, initial liquid temperature, and initial solid temperature—with each value being normalized. The tripartite presentation unveils the temporal ramifications on the neural network's efficiency, aligning with the elucidation provided in the Figure 10 analysis. Notably, at diminished time values, the error in images generated by the neural network becomes more accentuated. Furthermore, a discernible shift in the behavioral pattern of the neural network results over time is prominently observable and is particularly evident beyond the 220 s threshold.

An intriguing observation surfaces when considering the dispersion patterns over time for the three input parameters. Notably, it is the initial solid temperature variable that exhibits a more compacted distribution of data for each value. This phenomenon suggests that, despite encountering higher errors for these specific values, the neural network successfully captures the nuanced impact of all three characteristics. This analysis lends substantial weight to the theory articulated in the scrutiny of Figure 9. The proposition that network deficiencies may be rooted in the chosen dataset structure, where image changes predominantly manifest within the solid, gains further support. Additionally, it is plausible to assert that the temporal range employed in the simulation could contribute to idiosyncrasies in the resolution patterns assimilated by the neural network.

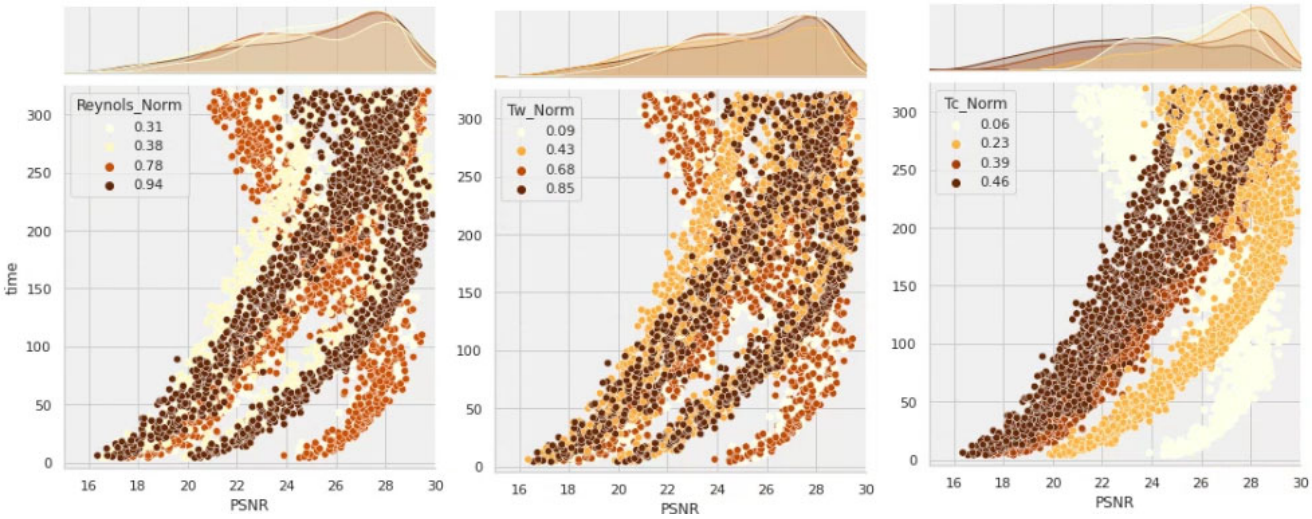
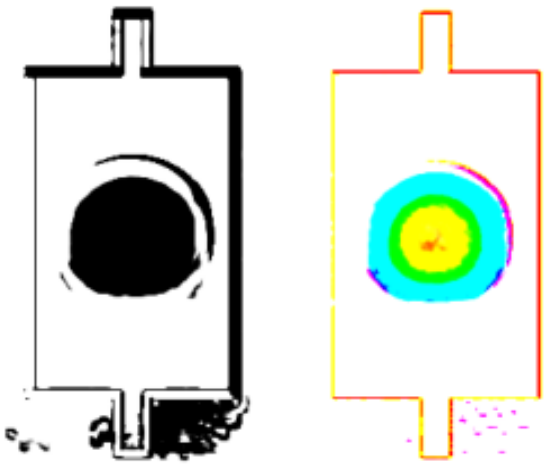


Figure 11. Dispersion of the generated images according to the PSNR parameter for the different types of variable features.

Despite the temporal effects on the noise in the generated images, it is noteworthy to observe, as depicted in Figure 12, that time does not significantly impact the structure of the images (represented by the SSI parameter). Additionally, one must acknowledge how the data in this parameter are clustered, exhibiting a behavior akin to the image noise parameter PSNR. Consequently, although time does not substantially affect the image structure, the error does increase for small time values, representing the forced state of the TES tank discharge system.



(**a**) (**b**)

Figure 12. Image of the TES tank used for this study. (**a**) Black shows the regions with errors across all of the images analyzed, and (**b**) color scale shows the regions with the highest concentrations of errors across the entire dataset analyzed.

The collective analysis of both Figures 11 and 13 reveals a uniform behavior pattern exhibited by the neural network concerning the input parameters explored in this study. However, the expansive ranges observed in the PSNR measurement prompt uncertainties about whether the current neural network model effectively assimilates these characteristics. The showcased impact of diverse time values on the network's accuracy serves as a poignant reminder of the imperative to consider the temporal variable in future developments. While the present study did not factor in time as a variable in the network training, acknowledging this facet may emerge as pivotal for mitigating errors in the neural network.

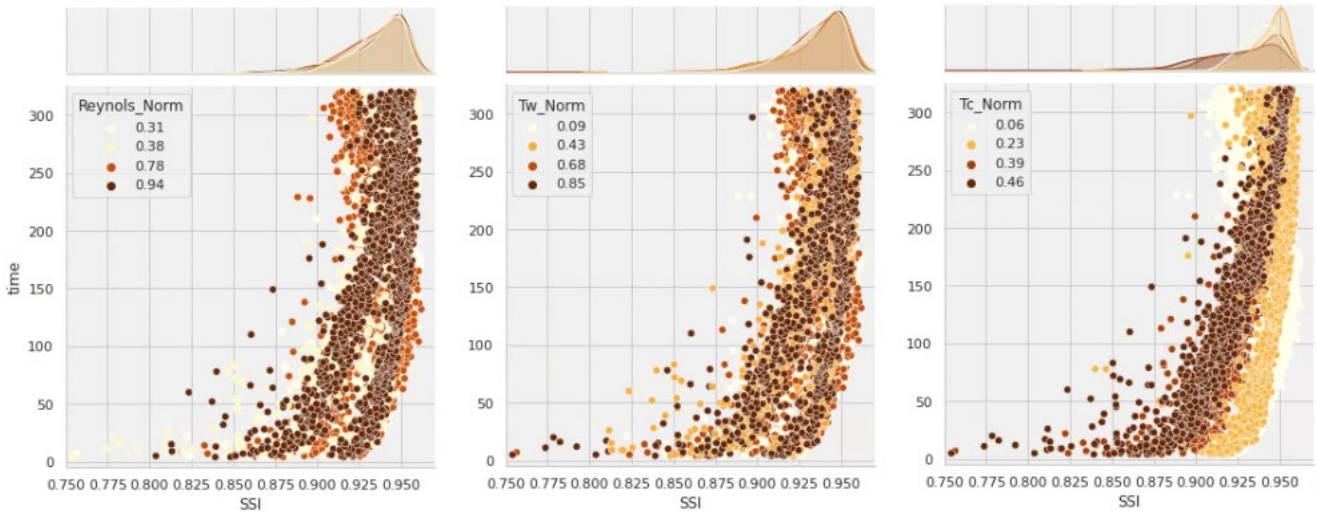


Figure 13. Dispersion of the generated images according to SSI parameters for the different types of variable features.

The latest analysis conducted adopts a graphical approach, intending to spotlight regions within the overall image of the TES tank that exhibit higher errors (pixels displaying more variation compared to the image generated via COMSOL). For this purpose, the difference between the images obtained from COMSOL and those generated by the neural network was computed, with the results represented on a gray scale. Subsequently, the 5072 obtained images were superimposed. Figure 12 encapsulates the outcomes, presenting the accumulated error regions in black on the right, while on the left, the regions with errors are depicted in a color scale, facilitating the visualization of areas where errors are more prevalent across the set of 5072 images.

In Figure 12a, an intriguing phenomenon comes to light: an accumulated error patch is evident in a region that is not part of the TES tank. This peculiarity arises due to the tank geometry, where COMSOL generates a white region that is disregarded in the computation by the software but not in the neural network inference. As this region remains consistent across all cases, the neural network tends to overfit, assigning pixel values that, while acceptable for the network, do not align with the expected pixel values. This phenomenon is perceptible in Figure 6, where all images exhibit a greyish tone in the aforementioned region.

The second region of interest in Figure 12a is the tank border. The entire contour exhibits distortions compared to the images generated in COMSOL. However, this error can be attributed to the fact that the current neural network is not designed to comprehend boundaries or regions, such as the void region and the TES tank region. This explanation can be extended to the region around the solid material contour in the tank. Due to the neural network's inability to comprehend environmental context, errors may arise in the space between the circle and the liquid material of the tank concerning images generated in COMSOL.

In Figure 12b, a color scale is employed to illustrate regions where errors appeared most frequently across the 5072 analyzed images. The prevalent red hue signifies areas with a higher recurrence of errors. In conjunction with the previously discussed issues regarding tank edges and the solid material circle, it is noteworthy to observe that, unlike the edges of the solid circle, the interior of the circle exhibits fewer errors per image than anticipated. This observation may bolster the hypothesis that a current challenge facing the neural network lies in its deficiency in interpreting contours or regions, given that most errors manifest at these boundaries.

4. Conclusions

The investigation endeavor has unveiled a substantial degree of congruence between many images generated by the neural network and those procured through COMSOL software. In the context of assessing how akin the images generated by the network are to those derived from CFD simulations, the neural network generates images with a high similarity index, above 0.9 for 94% of the test inputs, with the images generated by COMSOL. While this outcome may not attain the zenith of expectations within the artificial intelligence sphere, a similarity of such magnitude serves as a compelling testament to the prospective feasibility of integrating neural networks in TES tank modeling.

Despite the evident resemblance between the images generated by the neural network and their counterparts produced through COMSOL, the signal-to-noise ratio gives values that, although they can be considered acceptable, with a mean value 25 dB, indicate room for improvement. In other words, these generated images encompass pixel areas displaying colors that deviate from the expected norms. To ameliorate this aspect of the neural network-generated images, we plan to study further improvements to the architecture of the neural network and in the generation of the training dataset.

The architecture of the neural network employed in this study has proven to be a robust starting point for such analyses. Given the need to analyze not only an image but a series of values pertaining to the numerous characteristics of the tank and its materials, the architecture has demonstrated a commendable level of understanding and correlation with the variable features in the image. On the other hand, it has also highlighted the imperative for a more thorough analysis in selecting the dataset for training. This is crucial to mitigate the current errors present in the neural network.

In conclusion, addressing the overarching question framing this study, neural networks can simulate the behavior of a TES tank. However, this initial implementation of a neural network can and should be expanded to enable a comprehensive understanding of all of the relevant tank characteristics and materials used.

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

Funding: This project was funded by the European Union's Horizon 2020 Research and Innovation Programme under Grant No. 101007976 (CO-COOL). The views and opinions expressed are, however, those of the author(s) only and do not necessarily reflect those of the European Union or CINEA. Neither the European Union nor the granting authority can be held responsible for them. This work was partially funded by Ministerio de Ciencia e Innovación—Agencia Estatal de Investigación (AEI)—NextGeneration EU (TED2021-129462B-I00—MCIN/AEI/10.13039/501100011033/ NextGenerationEU/PRTR). This work was also partially funded by the Ministerio de Ciencia e Innovación—Agencia Estatal de Investigación (AEI) (PID2021-123511OB-C31—MCIN/AEI/10.13039/501100011033/ FEDER, UE, RED2022-134219-T, PID2022-139835NB-C22—MCIN/AEI/10.13039/501100011033/FEDER, EU, and PID 2023-152814OB-100—MCIN/AEI/10.13039/501100011033/FECER, EU). This paper is part of RYC2023-044196-I, funded by MCIU/AEI/10.13039/501100011033 and FSE+.

Data Availability Statement: Data are available upon request to the correspondence author.

Acknowledgments: The authors would also like to thank the Generalitat de Catalunya for the quality accreditation granted to their research group (2021 SGR 01615). GREiA is certified agent TECNIO in the category of technology developers of the Government of Catalonia. This work is partially

supported by ICREA under the ICREA Academia programme. Edgar F. Rojas Cala also is grateful to: "Convocatòria 2023 d'Ajuts UdL per la contractació de personal predoctoral en formació. Núm. expedient: 2023 UdL 12".

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

Acronym	Meaning
ANN	Artificial Neural Network
CFD	Computational Fluid Dynamics
CNN	Convolutional Neural Networks
DNN	Deconvolutional Networks
GAN	Generative Adversarial Network
MLP	Multilayer Perceptron
MLR	Multiple Linear Regression
PSNR	Peak Signal-to-Noise Ratio
PCM	Phase Change Material
RF	Random Forests
SSI	Structural Similarity Index
TES	Thermal Energy Storage
XGBoost	eXtreme Gradient Boosting

References

1. Cabeza, L.F.; Martorell, I.; Miró, L.; Fernández, A.I.; Barreneche, C. Introduction to thermal energy storage (TES) systems. In *Advances in Thermal Energy Storage Systems*; Elsevier: Amsterdam, The Netherlands, 2015; pp. 1–28. [CrossRef]
2. Arce, P.; Medrano, M.; Gil, A.; Oró, E.; Cabeza, L.F. Overview of thermal energy storage (TES) potential energy savings and climate change mitigation in Spain and Europe. *Appl. Energy* **2011**, *88*, 2764–2774. [CrossRef]
3. Alva, G.; Lin, Y.; Fang, G. An overview of thermal energy storage systems. *Energy* **2018**, *144*, 341–378. [CrossRef]
4. Mehling, H.; Cabeza, L.F. Heat and cold storage with PCM. In *An up to Date Introduction into Basics and Applications*, 1st ed.; Springer: Berlin/Heidelberg, Germany, 2008. [CrossRef]
5. Elalfy, D.A.; Gouda, E.; Kotb, M.F.; Bureš, V.; Sedhom, B.E. Comprehensive review of energy storage systems technologies, objectives, challenges, and future trends. *Energy Strategy Rev.* **2024**, *54*, 101482. [CrossRef]
6. Salman, R.; Aljabair, S. "Energy storage technology: The growing role of phase change materials in the construction industry" a review. *Int. J. Thermofluids* **2024**, *23*, 100712. [CrossRef]
7. Cabeza, L.F.; Cristian, S.o.l.é.; Castell, A.; Oro, E.; Gil, A. Review of Solar Thermal Storage Techniques and Associated Heat Transfer Technologies. *Proc. IEEE* **2012**, *100*, 525–538. [CrossRef]
8. Pompei, L.; Nardecchia, F.; Miliozzi, A. Current, Projected Performance and Costs of Thermal Energy Storage. *Processes* **2023**, *11*, 729. [CrossRef]
9. IEA. *Energy Transitions Require Innovation in Power System Planning*; IEA: Paris, France, 2022. Available online: https://www.iea.org/articles/energy-transitions-require-innovation-in-power-system-planning (accessed on 14 December 2023).
10. Ritchie, H.; Roser, M.; Rosado, P. Renewable Energy. Our World in Data. 2020. Available online: https://ourworldindata.org/grapher/solar-share-energy (accessed on 2 October 2023).
11. Cascetta, M.; Cau, G.; Puddu, P.; Serra, F. A comparison between CFD simulation and experimental investigation of a packed-bed thermal energy storage system. *Appl. Therm. Eng.* **2016**, *98*, 1263–1272. [CrossRef]
12. Schulte-Fischedick, J.; Tamme, R.; Herrmann, U. CFD Analysis of the Cool Down Behaviour of Molten Salt Thermal Storage Systems. In Proceedings of the ASME 2008 2nd International Conference on Energy Sustainability, Jacksonville, FL, USA, 10–14 August 2008; ASMEDC: Houston, TX, USA, 2008; Volume 2, pp. 515–524. [CrossRef]
13. Pandey, B.; Banerjee, R.; Sharma, A. Coupled EnergyPlus and CFD analysis of PCM for thermal management of buildings. *Energy Build.* **2021**, *231*, 110598. [CrossRef]
14. Kochkov, D.; Smith, J.A.; Alieva, A.; Wang, Q.; Brenner, M.P.; Hoyer, S. Machine learning–accelerated computational fluid dynamics. *Proc. Natl. Acad. Sci. USA* **2021**, *118*, e2101784118. [CrossRef]
15. Lira, J.O.B.; Riella, H.G.; Padoin, N.; Soares, C. Computational fluid dynamics (CFD), artificial neural network (ANN) and genetic algorithm (GA) as a hybrid method for the analysis and optimization of micro-photocatalytic reactors: NOx abatement as a case study. *Chem. Eng. J.* **2022**, *431*, 133771. [CrossRef]

16. Latest Neural Nets Solve World's Hardest Equations Faster Than Ever Before | Quanta Magazine. Available online: https://www.quantamagazine.org/latest-neural-nets-solve-worlds-hardest-equations-faster-than-ever-before-20210419/ (accessed on 16 May 2023).
17. Lu, L.; Jin, P.; Karniadakis, G.E. DeepONet: Learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators. *arXiv* **2019**, arXiv:1910.03193.
18. Chickerur, S.; Ashish, P. A Convolutional Neural Network Based Approach for Computational Fluid Dynamics. In Proceedings of the 2021 2nd International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE), Bengaluru, India, 16–17 December 2021. [CrossRef]
19. Usman, A.; Rafiq, M.; Saeed, M.; Nauman, A.; Almqvist, A.; Liwicki, M. Machine Learning Computational Fluid Dynamics. In Proceedings of the 2021 Swedish Artificial Intelligence Society Workshop (SAIS), Lulea, Sweden, 14–15 June 2021; pp. 1–4. [CrossRef]
20. Madhankumar, S.; Ilangovan, D.; Viswanathan, K. Integrating computational models and machine learning for corrugated absorber plate solar collector thermal predictions. *Process Saf. Environ. Prot.* **2024**, *188*, 336–349. [CrossRef]
21. Li, Y.; Huang, X.; Huang, X.; Gao, X.; Hu, R.; Yang, X.; He, Y.-L. Machine learning and multilayer perceptron enhanced CFD approach for improving design on latent heat storage tank. *Appl. Energy* **2023**, *347*, 121458. [CrossRef]
22. Kıyak, B.; Öztop, H.F.; Ertam, F.; Aksoy, I.G. An intelligent approach to investigate the effects of container orientation for PCM melting based on an XGBoost regression model. *Eng. Anal. Bound. Elements* **2024**, *161*, 202–213. [CrossRef]
23. Dhakane, V.; Mishra, P.; Yadav, A. Computational fluid dynamics (CFD)-deep neural network (DNN) model to predict hydrodynamic parameters in rectangular and cylindrical bubble columns. *Digit. Chem. Eng.* **2024**, *13*, 100185. [CrossRef]
24. Nadda, M.; Shah, S.K.; Roy, S.; Yadav, A. CFD-based deep neural networks (DNN) model for predicting the hydrodynamics of fluidized beds. *Digit. Chem. Eng.* **2023**, *8*, 100113. [CrossRef]
25. Rubio, G.; Valero, E.; Lanzan, S. Computational Fluid Dynamics Expert System using Artificial Neural Networks. *Int. J. Comput. Inf. Eng.* **2012**, *6*, 413–417.
26. Kukushkin, K.; Ryabov, Y.; Borovkov, A. Digital Twins: A Systematic Literature Review Based on Data Analysis and Topic Modeling. *Data* **2022**, *7*, 173. [CrossRef]
27. COMSOL-Software for Multiphysics Simulation. Available online: https://www.comsol.com/ (accessed on 16 May 2023).
28. Fastai-Welcome to Fastai. Available online: https://docs.fast.ai/ (accessed on 16 May 2023).
29. Welcome to Python.org. Available online: https://www.python.org/ (accessed on 16 May 2023).
30. PyTorch. Available online: https://pytorch.org/ (accessed on 18 December 2024).
31. Çengel, Y.A. *Heat and Mass Transfer: Fundamentals and Applications*, 5th ed.; SI Units; McGraw-Hill: New York, NY, USA, 2015.
32. Sobota, T. General Heat Conduction Equation in Various Coordinate Systems. In *Encyclopedia of Thermal Stresses*; Hetnarski, R.B., Ed.; Springer: Dordrecht, The Netherlands, 2014; pp. 1917–1926. [CrossRef]
33. Bourlard, H.; Kamp, Y. Auto-association by multilayer perceptrons and singular value decomposition. *Biol. Cybern.* **1988**, *59*, 291–294. [CrossRef]
34. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
35. Zeiler, M.D.; Krishnan, D.; Taylor, G.W.; Fergus, R. Deconvolutional networks. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; pp. 2528–2535. [CrossRef]
36. Fukushima, K. Neural network model for a mechanism of pattern recognition unaffected by shift in position-neocognitron. *Electron. Commun. Jpn.* **1979**, *62*, 11–18. [CrossRef] [PubMed]
37. Fukushima, K. Visual Feature Extraction by a Multilayered Network of Analog Threshold Elements. *IEEE Trans. Syst. Sci. Cybern.* **1969**, *5*, 322–333. [CrossRef]
38. Nair, V.; Hinton, G. Rectified Linear Units Improve Restricted Boltzmann Machines Vinod Nair. In Proceedings of the 27th International Conference on Machine Learning, Haifa, Israel, 21–24 June 2010; Volume 27.
39. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
40. Razavi-Far, R.; Ruiz-Garcia, A.; Palade, V. *An Introduction to Generative Adversarial Learning: Architectures and Applications*; Springer Science and Business Media Deutschland GmbH: Berlin/Heidelberg, Germany, 2022; Volume 217, pp. 1–6. [CrossRef]
41. Ganz, F.; Puschmann, D.; Barnaghi, P.; Carrez, F. A Practical Evaluation of Information Processing and Abstraction Techniques for the Internet of Things. *IEEE Internet Things J.* **2015**, *2*, 340–354. [CrossRef]
42. Manikandan, A.; Sanjay, T. *Generative Adversarial Networks: A Game Changer-GAN for Machine Learning and IoT Applications*; IGI Global: Hershey, PA, USA, 2023; pp. 67–83. [CrossRef]
43. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial networks. *Commun. ACM* **2020**, *63*, 139–144. [CrossRef]

44. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [CrossRef]
45. Huang, Z.; Jia, C.; Wang, S.; Ma, S. HMFVC: A Human-Machine Friendly Video Compression Scheme. *IEEE Trans. Circuits Syst. Video Technol.* **2022**. *early access*. [CrossRef]
46. Martini, M. A Simple Relationship Between SSIM and PSNR for DCT-Based Compressed Images and Video: SSIM as Content-Aware PSNR. In Proceedings of the 2023 IEEE 25th International Workshop on Multimedia Signal Processing (MMSP), Poitiers, France, 27–29 September 2023; pp. 1–5. [CrossRef]
47. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.