

Complete Deep Computer-Vision Methodology for Investigating Hydrodynamic Instabilities

Re'em Harel^{1,2}, Matan Rusanovsky^{2,3}, Yehonatan Fridman^{2,3}, Assaf Shimony⁴,
and Gal Oren^{3,4*}

¹ Department of Physics, Bar-Ilan University, Ramat-Gan, Israel

² Israel Atomic Energy Commission, P.O.B. 7061, Tel Aviv, Israel

³ Department of Computer Science, Ben-Gurion University of the Negev, P.O.B. 653,
Be'er-Sheva, Israel

⁴ Department of Physics, Nuclear Research Center-Negev, P.O.B. 9001, Be'er-Sheva,
Israel

reemharel22@gmail.com, matanru@post.bgu.ac.il, fridyeh@post.bgu.ac.il,
shimonya@gmail.com, orenw@post.bgu.ac.il

Abstract. In fluid dynamics, one of the most important research fields is hydrodynamic instabilities and their evolution in different flow regimes. The investigation of said instabilities is concerned with the highly non-linear dynamics. Currently, three main methods are used for understanding of such phenomenon – namely analytical models, experiments and simulations – and all of them are primarily investigated and correlated using human expertise. In this work we claim and demonstrate that a major portion of this research effort could and should be analysed using recent breakthrough advancements in the field of Computer Vision with Deep Learning (CVDL, or Deep Computer-Vision). Specifically, we target and evaluate specific state-of-the-art techniques – such as Image Retrieval, Template Matching, Parameters Regression and Spatiotemporal Prediction – for the quantitative and qualitative benefits they provide. In order to do so we focus in this research on one of the most representative instabilities, the Rayleigh-Taylor one, simulate its behaviour and create an open-sourced state-of-the-art annotated database (*RayleAI*). Finally, we use adjusted experimental results and novel physical loss methodologies to validate the correspondence of the predicted results to actual physical reality to prove the models efficiency. The techniques which were developed and proved in this work can be served as essential tools for physicists in the field of hydrodynamics for investigating a variety of physical systems, and also could be used via Transfer Learning to other instabilities research. A part of the techniques can be easily applied on already exist simulation results. All models as well as the data-set that was created for this work, are publicly available at: <https://github.com/scientific-computing-nrcn/SimulAI>.

Keywords: Fluid Dynamics, Hydrodynamic Instabilities, Rayleigh-Taylor Instability, Computer Vision, Deep Learning, Image Retrieval, Template Matching, Regressive Convolutional Neural Networks, Spatiotemporal Prediction

*Corresponding author

1 Introduction

The Rayleigh-Taylor instability (RTI) occurs in an interface between two fluids with different densities in which the lighter fluid pushes the heavier fluid [1]. RTI is found in many hydrodynamic experiments and natural phenomena such as Inertial Confinement Fusion (ICF), water suspended on oil in earth's gravity, astrophysical systems and many more [2]. Numerous experiments studying the growth of the instability and its effects on other phenomena are performed constantly all over the world [3–6] due to its importance. Generally speaking, there are two types of experimental platforms for investigating the evolution of RTI: Liquid or gas systems (for example [3–5]) and High-Energy-Density Physics (HEDP) systems, in which the fluids are in plasma state after being heated by powerful lasers (for example [7, 8]). In the former systems, it is difficult to control the initial perturbation but the time difference between consecutive frames is short comparing to the duration of the experiment, which typically can vary from milliseconds to seconds. Therefore, it is feasible to obtain with a fast camera tens or more frames per experimental shots. In the latter systems, the initial perturbation can be machined precisely prior to the laser drive, while the materials are in solid state, but the time scales are much shorter (about tens of nanoseconds) and only one or at most few frames can be obtained from a single experimental shot. Therefore, the experimental data from both types of experimental systems contain a partial reflection of the instability – either the exact initial perturbation or the detailed temporal evolution of the instability is missing, while both are crucial for understanding the phenomenon. The growth of the perturbation in RTI depends on numerous variables such as viscosity, ablation, surface tension, small density gradients and more. These variables are of different importance in different physical and experimental systems. In this work we consider the case of two incompressible and immiscible fluids and a single-mode sinusoidal initial perturbation. In this case, the early growth is exponential in time and is given by (via linear stability theory): $e^{\sqrt{\mathcal{A}kg}t}$ (1) where k is the wave number $k = \frac{2\pi}{\lambda}$, λ is the wavelength, g is the earth's gravity (and in the general case the acceleration of the system), and \mathcal{A} is the well known Atwood number, given by $\mathcal{A} = \frac{\rho_1 - \rho_2}{\rho_1 + \rho_2}$ (2) where ρ_1 is the density of heavier fluid and ρ_2 of the lighter one. In the late non-linear growth of such a single-mode perturbation, bubbles of the lighter fluid penetrate into the heavy fluid and spikes of heavy fluid penetrate into the light fluid in constant velocities, given by [9]: $u_{b/s} = \sqrt{\frac{2Ag\lambda}{c_d(1 \pm \mathcal{A})}}$ (3) where u_b and u_s are the velocities of the bubble and the spike, respectively. c_d is the drag coefficient, which equals to 6π and 3π for 2D and 3D, respectively. As the perturbation grows, the shear velocities on the sides of the bubbles and the spikes create vortices due to Kelvin-Helmholtz instability (KHI), in which the two materials mix in small length scales.

Needless to say that in reality (experiments), knowing the exact conditions of the density and the viscosity of the two fluids is unrealistic. A possible way to bridge this gap is via simulations [10] (which are much cheaper than performing additional experiments): Given the correct initial parameters one can simulate

the experiment and extract the missing time frames. However, initiating the simulation with the exact initial parameters is impossible due to the experimental uncertainties. Nevertheless, it is still a viable solution as one can run parameters-sweep and select the most similar simulation in comparison to the experiment. However, this solution might be difficult as there are many different parameters (both physical parameters with uncertainties and parameters in the analysis of the experimental results), which makes this process hard for a human. In the past years, the usage of different CVDL techniques in this scientific area is growing [11–19], and as it progresses, the above problem might also be solved using CVDL, since the introduction of CVDL techniques to the CFDs field proved to yield excellent results [20–26]. As different techniques in the CVDL field might be necessary in order to solve the problem, we define and devise several key problems, which collaboratively will enhance our understanding of RTI and other physical phenomena:

1. Given a diagnostic *image* from a simulation/experiment, *sort* a database in accordance with an image similarity score to the input.
2. Given a diagnostic *image* from simulation/experiment, *extract the parameters* of the simulation that yields a best match to an image in a database.
3. Given a *partial template* of a phenomena from a simulation/experiment, *find* matches in a database and *sort* them in accordance with an image similarity score to the input.
4. Given a *set of images* that correspond to a set of time steps, and a time parameter T , *predict* future non-existing time steps images.

#	Task	Quantitative & Qualitative Values
I.	Database sorting by an image similarity score.	<ul style="list-style-type: none"> • Meaningful order. • Extraction of non-regressive parameters.
II.	Regressive parameter extraction.	<ul style="list-style-type: none"> • Physical parameters that fit the experimental data. • Uncertainty margins of the model training.
III.	Find and sort partial templates by an image similarity score.	<ul style="list-style-type: none"> • Meaningful order. • A more extensive and more reliable matching survey which results in a decrease of the uncertainty margins for the template assumption comparing to analysis of the full images only.
IV.	Temporal inter/extrapolation of experiments.	<ul style="list-style-type: none"> • Data augmentation for low-data experiments. • Assurance and extension of a model.

Table 1: Quantitative and qualitative values for defined problems using CVDL.

The completion of the tasks above using CVDL techniques have both quantitative and qualitative values over classical optimization techniques, as presented in Table 1. In the first technique, a database (in our case, images) is being sorted by an image similarity score. When the similarity score is being weighted according to the physical significance of the features in the data, the resulted order is meaningful. In addition, the physical simulation parameters that yield the maximal image similarity (for example, with the experimental results) can be used

as a non-regressive optimization. Similarly, uncertainty margins for the physical parameters can be calculated by defining a minimal required similarity factor.

The second CVDL technique is an advanced optimization method: Given experimental results and corresponding simulations with a set of free parameters, the technique provides the values of the free parameters for a best fit to the experimental results *iff* the model training and validation loss convergence to approximately the same (small) value. The value of this method over the parameter extraction in the first technique is due to the regression process, which is significant when the simulation database is incomplete.

The third CVDL technique is similar to the first one, except for focusing on partial templates in the images instead of analyzing the whole image. It is valuable for physical problems in which there is a measurable pattern which is sensitive to the physical parameters. For example, in [27], the evolution of vortices, created by supersonic KHI, was measured and compared with hydrodynamic simulations. The analysis in [27] was based on the large-scale structures (i.e. the widths of the vortices). The medium-scale structures (i.e. the roll-ups within the vortices) were measured in the experiment and were compared to the simulations qualitatively. A more detailed analysis of the templates of the roll-ups could provide additional physical insights such as the effect of viscosity in the experimental conditions. Another example, which is relevant to the evolution of RTI, is originated from morphology differences between experiments in HEDP platforms and simulations of them. A detailed analysis of the morphology of the bubbles and the spikes can provide insights on magnetic effects due to the plasma conditions in these experiments [28] or perhaps other physical effects. A third example is the measurement of ablative RTI [29, 30] which is relevant to astrophysical systems. Similarly to the KHI example, The ablation effects were analyzed by the width of the mixing zone. The morphology of the spikes was affected by the ablation as clearly seen from the experimental images and simulations. A detailed analysis of the structure of the spikes can provide further insights on the ablation effects. In each of the examples above, the CVDL technique would provide a meaningful image similarity order between the sub-figures within the simulation images to the template input of the experimental image. In addition, this technique provides a more extensive and more reliable matching survey, which can decrease the uncertainty margins for the physical parameters, comparing to analysis of the full images only. Therefore, it can serve as a convenient method for analyzing the physical effects and their significance.

The fourth CVDL technique provides a temporal interpolation/extrapolation of experimental results. First, it can provide data augmentation for low-data experiments. Moreover, it can be useful especially when there is a difference between the outputs of the simulations and the experimental images (for example, when the simulation covers a part of the experimental platform) and a prediction of experimental results at additional times are needed. This technique can also be used for an assurance of the model: When there is a series of experimental images, the training can be performed on a part of them. When the training process is accomplished, the prediction of the model for the times of the images

that were not selected for the training can be compared for the experimental images from that times. The similarity score between the predicted images and the experimental images can be used for the validity of the model.

A useful feature of the techniques above is that they can be applied on existing simulations databases, since physicists usually perform parameter surveys when analyzing experimental data using simulations. In other words, it can be utilized on previous works without running any additional simulation.

2 *RayleAI* – Database Characteristics

In order to implement the CVDL techniques described above, we first present a state-of-the-art annotated database named *RayleAI* [31], which contains thresholded images from a simulation (using the DAFNA hydrodynamic code [32]) of a sinusoidal single-mode RTI perturbation, given by $y = h \cos\left(\frac{2\pi x}{\lambda}\right)$, with a resolution of 64x128 cells, 2.7cm in x axis and 5.4cm in y axis, while each fluid follows the equation of state of an ideal gas (with adiabatic index $\gamma = \frac{5}{3}$) and a hydrostatic equilibrium was set adiabatically with a pressure of 1 bar on the interface. The simulation input consists of three free parameters: Atwood number, gravitational acceleration and the amplitude of the perturbation (as well as additional time parameter). The database contains 101,250 images produced by 1350 different simulations (75 time steps each) with unique set of the free parameters per each simulation. The format of the repository is built upon directories, each represents a simulation performance with the directory name indicating the parameters of the specific simulation. For example, the directory *gravity_750_amplitude_0.5_atwood_0.16* is a simulation with $g = 750 \frac{\text{cm}}{\text{s}^2}$, initial amplitude of 0.5, and $\mathcal{A} = 0.16$. The ranges of the Atwood number, gravity and initial perturbation are presented in Table 3. Table 2 shows the simulation images from DAFNA compared to the experimental images. The simulation images presented are with $g = 750 \frac{\text{cm}}{\text{s}^2}$ and $\mathcal{A} = 0.16$.

The choice of these exact parameters was derived from well known experimental results [6]. The physical parameters in the experiment were Atwood of $\mathcal{A} = 0.155$, gravity of $g = 740 \frac{\text{cm}}{\text{s}^2}$ and the initial perturbation wavelength of 0.54cm. The initial amplitude is not given but can be estimated from the first image by about 0.1cm. Thus, one can deduce that the simulation in the database with $\mathcal{A} = 0.16$, $g = 750 \frac{\text{cm}}{\text{s}^2}$ and $h = 0.1$ should produce the most accurate result match (as shown in Table 2). An example of an experimental image is shown in Fig. 1. The experiment images were originally taken in grayscale. For optimal results, each image was processed with two methods (Erode-Dilate vs. Histogram Equalization), and the most fitting result that resembles the interface between the two fluids best (by expert opinion) was selected, cropped and resized, and then binarized by a threshold. Those images are also included in the database under */experiment*.










$\mathbf{h} \backslash \mathbf{T}$	0.2	0.3	0.4
Exp'			
0.1			
0.2			

Table 2: Diagnostic of RTI in different T and h values from *RayleAI*.

Parameter	From	To	Stride
Atwood (\mathcal{A})	0.02	0.5	0.02
Gravity (g) [cm/s^2]	600	800	25
Amplitude (h) [cm]	0.1	0.5	0.1
X [cm]	2.7	2.7	0.0
Y [cm]	5.4	5.4	0.0

Table 3: Simulation parameters.

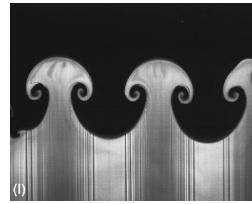


Fig. 1: The full image from the experiment ($T=0.4\text{s}$).

3 Deep Computer-Vision Methods

3.1 Task I: Image Retrieval using InfoGAN

Generative Adversarial Networks (GANs) [33] is a framework capable to learn a *generator* network G , that transforms noise variable z from some noise distribution into a generated sample $G(z)$, while the training of the generator is optimized against a *discriminator* network D , which targets to distinguish between real samples with generated ones. The fruitful competition of both G and D , in the form of MinMax game, allows G to generate samples such that D will have difficulty with distinguishing real samples between them. The ability to generate indistinguishable new data in an unsupervised manner is one example of a machine learning approach that is able to understand an underlying deep, abstract and generative representation of the data. Information Maximizing Generative Adversarial Network (InfoGAN) [34] utilizes latent code variables c_i , which are added to the noise variable. These noise variables are randomly generated from a user-specified domain. The latent variables impose an Information Theory Regularization term to the optimization problem, which forces G to preserve the information stored in c_i through the generation process. This allows learning interpretative and meaningful representations of the data, with a negligible computation cost, on top of a GAN. The high-abstract-level representation can be extracted from the discriminator (e.g. the last layer before the classification) into a features vector. We use these features in order to measure the similarity between some input image to any other image, by applying some distance function (e.g. l_2 norm) on the features of the input to the features of the

other image. This methodology provides the ability to order images similarity to a given input image [35].

In order to evaluate InfoGAN performances over *RayleAI*, we also use – for comparison – the Computer-Vision technique of LIRE [36]. LIRE is a library that provides image retrieval from databases based on image characteristics among other classic features. LIRE creates a *Lucene* index of image features using both local and global methods. For the evaluation of the similarity of two images, one can calculate their distance in the space they were indexed to. Many state-of-the-art methods for extracting features can be used, such as Gabor Texture Features [37], Tamura Features [38], or FCTH [39]. For our purposes, we found that Tamura Features method is the most suitable method that LIRE provides, as it indexes *RayleAI* images in a more dispersed fashion. The Tamura feature vector of an image is an 18 double values descriptor that represents texture features in the image that correspond to human visual perception.

3.2 Task II: Parameters Regression using ConvNet – pReg

Many Deep Learning techniques obtain state-of-the-art results for regression tasks, in a wide range of CV applications [40] such as Pose Estimation, Facial Landmark Detection, Age Estimation, Image Registration and Image Orientation [41] [42]. Most of the deep learning architectures used for regression tasks on images are Convolutional Neural Networks (ConvNets), which are usually composed of blocks of Convolutional layers followed by a Pooling layer, and finally Fully-Connected layers. The dimension of the output layer depends on the task, and its activation function is usually linear or sigmoid.

ConvNets can be used for retrieving the parameters of an experiment image, via regression. Our model (henceforth *pReg*) (Fig. 3) consists of 3 Convolutional layers with 64 filters, with a kernel size 5×5 , and with l_2 regularization, each followed by a Max-Pooling layer, a Dropout of 0.1 rate and finally Batch Normalization. Then, there are two Fully-Connected layers of 250 and 200 features, which are separated again by a Batch Normalization layer. Finally, the Output layer of our network has 2 features (as will be described next), and is activated by sigmoid to prevent the exploding gradients problem. Since the most significant parameters for describing each image frame are Amplitude and Time – which *pReg* is trained to predict – we used only a subset of *RayleAI* for the training set, namely images with the following parameters: $\mathcal{A} \in [0.08, 0.5]$ (with a stride of 0.02), $g \in \{625, 700, 750, 800\}$, $h \in [0.1, 0.5]$ (with a stride of 0.1) and $T \in [0.1, 0.6]$ (with a stride of 0.01). We fixed a small amount of values for Gravity and for Amplitude, so the network will not try to learn the variance that these parameters impose while expanding our database with as minimal noise as possible. We chose the value ranges of Atwood and Time in order to expose the model to images with both small and big perturbations, such that the amount of the latter ones will not be negligible. Our reduced training set consists of $\sim 16K$ images, and our validation set consists of $\sim 4K$ images. Nonetheless, for increasing generalization and for decreasing model overfitting, we employed data augmentation. Since there is high significance for the perspective from which each image

is taken, the methods of data augmentation should be carefully chosen: Rotation, shifting and flipping methods may generate images such that the labels of the original parameters do not fit for them. Therefore, we augment our training set with only zooming in/out (zoom range=0.1) via TensorFlow [43] preprocessing.

3.3 Task III: Quality-Aware Template Matching using QATM

One variation of the Template Matching problem is defined as follows: Given an exemplar image E , find the most similar region of interest in a target image S [44]. Classic template matching methods often use Sum-of-Squared Differences (SSD) or Normalized Cross-Correlation (NCC) to assess the similarity score between a template and an underlying image. These approaches work well when the transformation between the template and the target search image is simple. However, with non-rigid transformations, which are common in real-life, they start to fail. Quality-Aware Template Matching (QATM) [45] method is a standalone template matching algorithm and a trainable layer with trainable parameters that can be embedded into any Deep Neural Network. QATM is inspired by assessing the matching quality of the source and target templates. It defines the $QATM(e, s)$ -measure as the product of likelihoods that a patch s in S is matched in E and a patch e in E is matched in S . Once $QATM(e, s)$ is computed, we can compute the template matching map for the template image E and the target searched image S . Eventually, we can find the best-matched region R^* which maximizes the overall matching quality. Therefore, the technique is of great need when templates are complicated and targets are noisy. Thus most suitable for RTI images from simulations and experiments.

3.4 Task IV: Time Series Prediction using PredRNN

Learning the evolution of the RTI in order to predict future time or gap frames requires both understanding of the spatial aspects of each time frame (e.g. the interface between the fluids), and understanding of time development: As the time progresses, the simulation tends to be more and more chaotic. Convolutional Long Short Term Memory networks (CLSTMs) [46] is a class of algorithms which are able to predict future image states by past and present image states based on training sequences of images. The architecture of this network is based on a two-dimensional grid of units that pass spatial information vertically (upwards), and temporal information horizontally (rightwards). However, the standard CLSTMs architectures lack the capability of preserving the temporal information for long terms, since the spatial information that is learned via the top unit in a specific time step, is not passed to the bottom unit in the next time step, leading to the loss of important information. PredRNN [47] is a state-of-the-art Recurrent Neural Network for predictive learning using LSTMs. PredRNN memorizes both spatial appearances and temporal variations in a unified memory pool. Unlike standard LSTMs, and in addition to the standard memory transition within them, memory in PredRNN can travel through the whole network in a zigzag direction, therefore from the top unit of some time step to the bottom unit of

the other. Thus, PredRNN is able to preserve the temporal as well as the spatial memory for long-term motions. In this work, we use PredRNN for predicting future time steps of simulations as well as experiments, based on the given sequence of time steps.

4 Evaluation Methodology

In order to test how the discussed above techniques perform on physical simulations as well as experiments, we propose new task-specific test methods, that quantify how well each technique operates on a concrete database. We present novel evaluation methodologies for the techniques presented in Sections 3.1, 3.2 and 3.3, based on a suitable corresponding loss measure for the first two tasks, and a sophisticated clustering-visualization method for the third. The evaluation of last forth technique (3.4) will be discussed separately.

The first evaluation method, namely *Physical Loss*, quantifies *how meaningful the results of the technique are*, i.e. whether the results of the technique are reflected in the (physical) annotations of the data. For example, in the case of Image Retrieval, it is inconclusive to decide whether the results are sufficient solely based on visual examination, since it is a very difficult task for humans to determine the correct ordering of lots of results, thus it is hard to establish whether the technique is satisfactory. Therefore, we suggest to measure how each input image is *physically close* to each of the returned image outputs, based on some or all of their parameters labels. Thus, for Image Retrieval (and for Parameters Regression, explained later in Section 5.2), each output image gets two scores – one from the technique at hand, e.g. similarity score, and one from the difference between its parameters to the parameters of the input image. In cases of high correlations of these scores, one can infer that indeed the results of the technique are of a meaningful (physical) value. We chose the Mean Squared Error (MSE) as the parameter difference function, although it may be calculated via any desired error function. We note that since the ranges of the parameters are scaled differently, we suggest to normalize them beforehand.

Furthermore, one important aspect that results from *Physical Loss* is the ability to identify the parameters which are likely to produce a small impact on the simulation results (depending on time). For example, in the case of small ratio between the amplitude and the wavelength of the perturbation (up to a few percent), RTI grows linearly according to Eq. 1 and approximately preserves its initial shape. Therefore, two simulations that differ only by their initial small amplitudes will practically result in the same late evolution up to a constant time shift. As a result, it is expected from physical considerations that if one produces an amplitude-based *Physical Loss* methodology for late times, the CVDL techniques will generate semi-random values of error as the amplitude hardly affect the simulation in late times. A similar result is also expected for the gravity parameter since for incompressible fluids (a good approximation in our case), two simulations that differ only by the gravity parameter will practically result in the same evolution as a function of the normalized time. A useful definition

of the normalized time is $\tilde{t} = \sqrt{\frac{Ag}{\lambda}}t$ as also reflected from Eq. 1. Concluding the above physical influence of the initial amplitude and the gravity parameters, only the Atwood and time parameters should have a significant impact on the results and are expected to be identified using the physical loss methodology.

In cases where there are no meaningful (physical) annotations, we developed another new evaluation method. Specifically in the case of Template Matching, where some partial template is searched through a database. Unlike the physical loss case, the physical parameters of the returned partial region of interest have no unique physical labels, since we might expect to find this template in images from a wide range of different parameters. For that end, we present a relaxed-evaluation method, that quantifies how well the technique at hand separates similar images from dissimilar images. Similarly to our *Physical Loss* methodology, we use two values for each output: The score from the technique, and a cluster number – returned from some unsupervised clustering algorithm. Scenarios in which continuous sequences from the results of the technique are from the same cluster might indicate the ability of the technique to perform a proper distinction between classes of similarity to a given input template. Alternatively, cases of sequences of results from mixed clusters, especially in the first and most similar regions, might prove that the technique did not succeed in separating the most similar images from the rest. Accordingly, we applied K-Means as our clustering algorithm, after extracting the main features from each image through Principal Component Analysis (PCA) to achieve more precise results [48]. Next, we present in Section 5 the evaluation results of the techniques from Section 3.

5 Results and Discussion

5.1 Task I: Image Retrieval

In order to test the performance of InfoGAN against LIRE we chose two separate test cases. In the first general test case, we chose 13,000 random input test images from *RayleAI*. In the latter, we chose approximately 10,000 input images with $T > 0.25$ in order to pick the most *complex* images, as the RTI is more chaotic and dominant in this regime. We then executed InfoGAN and LIRE on the entire *RayleAI* data-set for both test cases. Then, for each tested image and for each tool, we sorted the results according to the similarity scores that were given by the model. To evaluate the results and quantify how well the tool performed, we employed the *physical loss* methodology, introduced in section 4, over the Atwood parameter. Then, for each tool and test case, we calculated the average physical loss.

In Fig. 2 we present the physical loss methodology (using a comparison between the technique score [in blue] and the physical loss [in red] per each index, and draw a thin blue line to correlate them) only on the Atwood parameter, as it is the most significant parameter. In Figures 2a and 2b, we observe that InfoGAN outperformed LIRE on the complex images test, as the averaged physical loss of the first – and most important – indices of InfoGAN is ~ 0.25 , in contrast

to ~ 0.4 of LIRE. Furthermore, InfoGAN outperforms LIRE along the entire 2000 first examined indices, showing many powerful capabilities in the complex data case. In Figures 2c and 2d, we can see that InfoGAN and LIRE perform quite the same in the first indices, with averaged physical loss of around 0.4. Yet, if we focus on the entire 2000 first indices, we see that InfoGAN starts to outperform LIRE with smaller physical loss values. Additionally, it seems that there is a higher correlation between the scores of InfoGAN to their corresponding physical loss values (blue and red lines act accordingly) in each of the test cases, which indicated again on the ability of InfoGAN to learn the underlying physical pattern of the data. Another important aspect in which InfoGAN outperforms LIRE in both test-cases is the width of the physical loss line: As the red line is thinner, there is less noise and therefore the results have more physical sense. Although it seems that all red lines are of approximately the same width, the lines of InfoGAN are much thinner since the score ranges of InfoGAN are smaller than the ones of LIRE (scaled from 0 to 1, in contrast to LIRE which are scaled from 0 to 1.4). Note, that in all Figures, the blue lines are normalized by the min-max normalization method, contrary to the red lines which are presented as raw values. The overwhelming superiority of InfoGAN is somehow expected and can be explained as the ability of a deep learning model to learn complex patterns from our tailor-made and trained database. However, although LIRE provides decent results without requiring to be trained on a specific organized database (which obtaining is not always an easy task), it is still a classic image processing tool, which lacks the learning capabilities that will show it to understand deep patterns from the data (exemplification in Table 4). Therefore, for image retrieval applications with suitable databases, we suggest applying InfoGAN.

















































Index	1	2	3	4	5	6	7	8	9	10	11	12
LIRE												
												
InfoGAN												
												

Table 4: LIRE and InfoGAN first 12 results for two chosen test images.

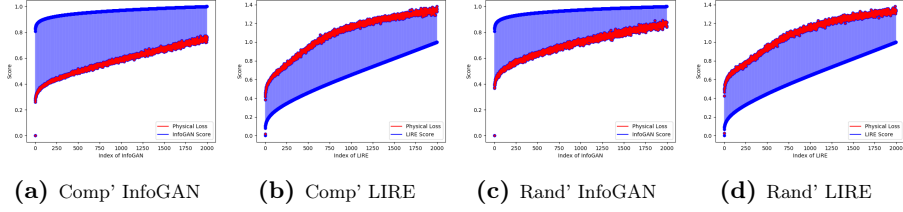


Fig. 2: LIRE and InfoGAN averaged *physical loss* methodology over Atwood.

5.2 Task II: Parameters Regression

In order to test the performances of our *pReg* network, we employed evaluation tests that are similar to the tests presented in section 4. We used *pReg* to predict the – activated by sigmoid – parameters: \mathcal{A} and T for 2,000 random images. Then, for each image we searched through *RayleAI* for the 2,000 images with the lowest scores, based on their l_2 distance between their \mathcal{A} and T activated by sigmoid parameters, to that of the input image.

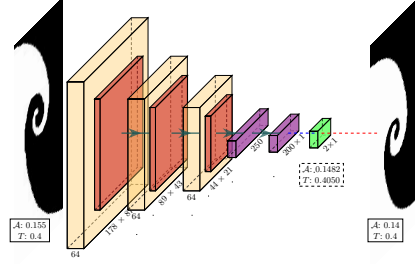


Fig. 3: CNN model, named *pReg*, for \mathcal{A} and T parameters Regression. The left image is an example of an experimental input, with the real parameters of \mathcal{A} : 0.155, T : 0.4, which the model predicts for the values \mathcal{A} : 0.1482, T : 0.4050 as can be seen under the green layer. The red dotted line indicates the similarity search operation that quantifies the distance of images from *RayleAI*, based on the l_2 distance over \mathcal{A} and T .

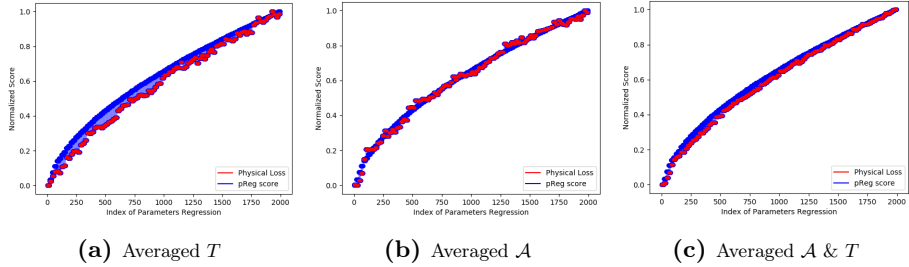


Fig. 4: *pReg* averaged *physical loss* methodology over Atwood and time.

















$g:-800,h:0.5,$ $\mathcal{A}:0.34,T:0.37$	$g:-625,h:0.5,$ $\mathcal{A}:0.34,T:0.36$	$g:-800,h:0.2,$ $\mathcal{A}:0.34,T:0.36$	$g:-800,h:0.3,$ $\mathcal{A}:0.34,T:0.36$
			
input	1	2	3
$g:-700,h:0.3,$ $\mathcal{A}:0.34,T:0.36$	$g:-625,h:0.1,$ $\mathcal{A}:0.34,T:0.36$	$g:-750,h:0.5,$ $\mathcal{A}:0.34,T:0.36$	$g:-750,h:0.2,$ $\mathcal{A}:0.34,T:0.36$
			
4	5	6	7
$g:-750,h:0.1,$ $\mathcal{A}:0.34,T:0.36$	$g:-625,h:0.3,$ $\mathcal{A}:0.34,T:0.36$	$g:-800,h:0.4,$ $\mathcal{A}:0.34,T:0.36$	$g:-750,h:0.3,$ $\mathcal{A}:0.34,T:0.36$
			
8	9	10	11
$g:-700,h:0.5,$ $\mathcal{A}:0.34,T:0.36$	$g:-700,h:0.2,$ $\mathcal{A}:0.34,T:0.36$	$g:-750,h:0.4,$ $\mathcal{A}:0.34,T:0.36$	$g:-625,h:0.4,$ $\mathcal{A}:0.34,T:0.36$
			
12	13	14	15

Table 5: pReg Similarity based on parameters regression.

As can be seen in Fig. 4a, the model explains very well the time, especially in the lowest (≤ 500) and highest (≥ 1800) indices, since the red dots of the normalized *physical loss* over the time, and the blue dots of the normalized l_2 distance from the predicted parameters, act similarly. The higher difference between the dots in the middle of the scale ($500 < \text{indices} < 1800$) is somehow expected, as it is harder for models to predict the parameters accurately in cases where there is a 'mild' physical difference. Yet, in cases where there is a high resemblance or significant difference with respect to the physical loss, it is more likely that the model will predict similar parameters or very different parameters, respectively. In Fig. 4b, we can see that the model explains even better the Atwood parameter, as the graphs are almost the same with some small noise. This can be explained by the significance and importance of the Atwood parameter. In Fig. 4c we see that the combination of Atwood and time greatly outperforms the former two cases, since the red line almost converges to the blue line. We note that as the predicted parameters in *pReg* are only \mathcal{A} and

T , and the difference is calculated only over them, for each input image there are lots of images in *RayleAI* with the same calculated distance – same \mathcal{A} and T , but different g or h . Therefore the trends in Figs. 4b, 4a, 4c might have dense blocks of dots – with same or very similar scores. Furthermore, the fact that there are lots of images with the same loss results in highly similar median and average values trends, therefore we present only the average values graphs from space considerations. Finally, since images with the same \mathcal{A} and T but different g or h have the same loss (over \mathcal{A} and T), they are ordered arbitrarily. Therefore, the physical loss over all parameters does not explain the similarity order of $pReg$, because of the arbitrariness that g and h impose. In Table 5 we present an example for image similarity search based on physical parameters regression.

5.3 Task III: Template Matching

For the evaluation of the QATM algorithm we cropped 16 templates from the experiment images. For each template, we employed the following procedure: We ran the QATM algorithm (1-to-1 version, the most precised one) on each image in *RayleAI* and found a matched sub-figure. Then, we sorted the results in accordance to the QATM similarity scores. For results evaluation, we employed the loss methodology of PCA and K-Means, as described in Section 4.

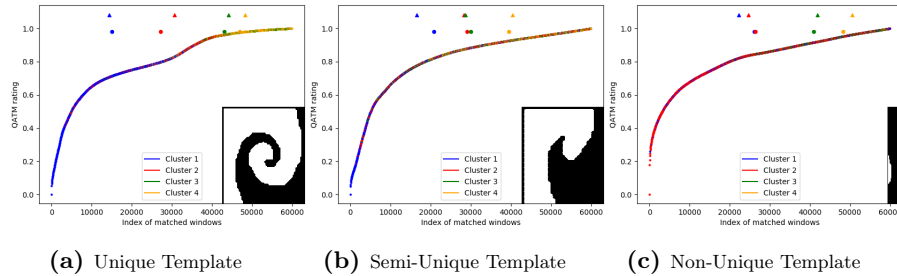


Fig. 5: PCA and k-means clustering methodology made on QATM results.

In Fig. 5 we present the results of three representative templates, while in each the normalized score results of the algorithm are sorted in an increasing order. The color of every point represents the cluster of the underlying sub-figure, returned by the K-Means algorithm. The up-arrows and the circles lying above the curves represent the median and the average of the indices of each cluster, respectively. To keep the trends readable, only one of each 30 dots is presented. As can be seen in Fig. 5a, the clustering algorithm divides the indices into 4 separate and distinct areas: There is pure congestion of blue dots in the first thousands of indices, without any rogue non-blue dots. This indicates that the algorithm understood the template successfully and found lots of significantly similar sub-figures. This powerful result might be explained because the searched

template is of a 'unique' shape, which helps QATM extract lots of features and compare them to *RayleAI*. in Fig. 5b, we can see pure congestion of blue dots in the first hundreds of indices, and a mixture of blue and red dots, with the unignorable presence of green dots in the right following indices. This mixture of clusters, that appears in relatively small indices, indicates that the algorithm's results start to be less meaningful after a couple of hundreds of indices. This can be explained since the searched template is less unique than the previous template. Finally, in Fig. 5c the blue and red clusters seem to be inseparable all along the index axis. This indicates that the algorithm did not understand well the template and has difficulties to bring quality matched sub-figures, as expected from the lack of uniqueness of the template. In Table 6 we present first raw results of said tests.










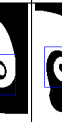





























Index	1	2	3	4	5	6	7	8	9	10	11	12
 Unique												
 Semi-Unique												
 Non-Unique												

Table 6: QATM best matchings for the templates discussed in Fig. 5.

5.4 Task IV: Spatiotemporal Prediction

The PredRNN model was trained on *RayleAI* sequences of 0.01s time steps. As mentioned, the RTI experiment with the parameters $g = 740$, estimated amplitude, and $\mathcal{A} = 0.155$ contains 12 black and white images with an interval of ~ 0.033 s between each couple of consecutive frames, while the time steps of our simulation are of 0.01s. In order to fill the missing time steps, we used PredRNN to predict the missing time intervals of the experiment. We filled the missing time steps in an iterative manner, by predicting a single future time step at a time. Furthermore, we tested the quality of the prediction of a simulation with the following parameters: $g = 725$, $\mathcal{A} = 0.14$ and $h = 0.3$. As an input, the first 10 images of the simulation were given, while predicting a total of 49 time steps.

















































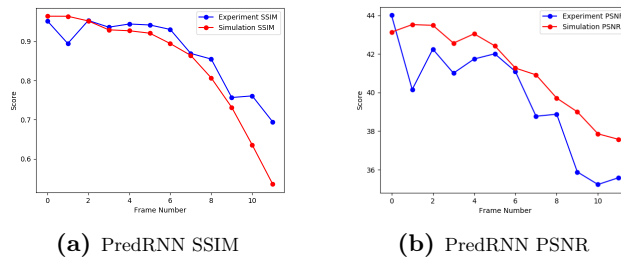
Time	0.03	0.06	0.1	0.13	0.16	0.2	0.23	0.26	0.3	0.33	0.36	0.4
Exp'												
Pred'												
Sim'												
Pred'												

Table 7: PredRNN prediction of the experiment and simulation.

The results of PredRNN prediction are shown in Table 7. The columns represent time steps ranging from $t_{init} = 0.03$ to $t_{fin} = 0.4$. The first and third rows of images represent the images of the corresponding time steps of the experiment and the simulation respectively, and as such consider to be GTs. The second and the fourth rows of images represent the prediction of PredRNN on the corresponding time steps of the *filled* experiment and the simulation respectively. As one can see, PredRNN produces very similar predictions to the GT images. In Fig. 6 we quantify the quality of the predicted images using PSNR and SSIM evaluation tests between the produced PredRNN images to their corresponding GT images, similarly to [47]. Both scores measure the similarity between the predicted frame and its corresponding GT frame. These scores decrease as time progresses, due to the expected difficulty of the model to predict the distant future. However, it's worth noting that a simple image sharpening on the predicted results can dramatically increase both SSIM and PSNR scores.

**Fig. 6:** SSIM and PSNR scores of the predicted experimental and simulated results.

6 Conclusions and Future Work

In this work we presented our state-of-the-art complete CVDL methodology for investigating hydrodynamic instabilities. First, we defined the problems and emphasised their significance. Second, we created a new comprehensive tagged database for the needed learning process, which contains simulated diagnostics for training, and experimental ones for testing. Third, we showed how our methodology targets the main acute problems in which CVDL can aid in the current analysis process, namely using deep image retrieval; regressive deep convolutional neural networks; quality aware deep template matching; and deep spatiotemporal prediction. Fourth, we formed a new physical loss and evaluation methodology, which enables to compare the performances of the model against the physical reality, and by such to validate its predictions. At last, we exemplified the usage of the methods on the trained models and assured their performances using our physical loss methodology. In all of the four tasks, we managed to achieve excellent results, which prove the methodology suitability to the problem domain. Thus, we stress that the proposed methodology can and should be an essential part of the hydrodynamic instabilities investigation toolkit, along with analytical models, experiments and simulations.

In regard to future work, an extension of the methodology might be useful for solving the discrepancy between simulations and experiments when it is clear that the initial parameters of the simulation is not covering the *entire* physical scope. Since in many cases a classical parameter sweep does not yield the desired results, an extension of the model – in the form of unmodelled parameters regression – should be used. For example, it might be useful for the physical problem presented in [29, 30], in which the simulation results do not cope with the experimental ones. Thus, an extended deep regressive parameter extraction model should be applied in a new form such that unknown parameters – i.e. parameters which are not part of the simulation initiation – could be discovered and formulated. This is crucial, as numerous current efforts suggest that often there is a missing part in the understanding of the simulated results. Thus, preventing any traditional method to match the simulated results to the experimental ones. Once discovered, in order to understand and formulate said unknown parameters, an extensive Explainable AI (XAI) methodology [49] should be performed.

Another strength of the presented methodology is that it can be applied on an already existing data in case that parameter sweep was previously performed on other physical data. Therefore, it might yield physical insights without running any additional simulation. In addition, the toolkit can be easily suited to the physical problem. For example, if the width of vortices was investigated in a previous research [27, 50, 51], template matching would be useful for investigating the inner structure of the vortices; With series of experimental images from different times at hand, the spatiotemporal prediction can be used for prediction of results at unmeasured times. Finally, the models presented in this work might be invaluable for learning physical problems with less training data and a more complex form, using Transfer Learning [52].

Acknowledgments

This work was supported by the Lynn and William Frankel Center for Computer Science. Computational support was provided by the NegevHPC project [53].

References

1. David H Sharp. Overview of rayleigh-taylor instability. Technical report, Los Alamos National Lab., NM (USA), 1983.
2. Philip G Drazin. *Introduction to hydrodynamic stability*, volume 32. Cambridge university press, 2002.
3. KI Read. Experimental investigation of turbulent mixing by rayleigh-taylor instability. *Physica D Nonlinear Phenomena*, 12:45–58, 1984.
4. Stuart B Dalziel. Rayleigh-taylor instability: experiments with image analysis. *Dynamics of Atmospheres and Oceans*, 20(1-2):127–153, 1993.
5. Guy Dimonte and Marilyn Schneider. Turbulent rayleigh-taylor instability experiments with variable acceleration. *Physical review E*, 54(4):3740, 1996.
6. JT Waddell, CE Niederhaus, and Jeffrey W Jacobs. Experimental study of rayleigh-taylor instability: low atwood number liquid systems with single-mode initial perturbations. *Physics of Fluids*, 13(5):1263–1273, 2001.
7. JP Knauer, R Betti, DK Bradley, TR Boehly, TJB Collins, VN Goncharov, PW McKenty, DD Meyerhofer, VA Smalyuk, CP Verdon, et al. Single-mode, rayleigh-taylor growth-rate measurements on the omega laser system. *Physics of Plasmas*, 7(1):338–345, 2000.
8. Bruce A Remington, Hye-Sook Park, Daniel T Casey, Robert M Cavallo, Daniel S Clark, Channing M Huntington, Carolyn C Kuranz, Aaron R Miles, Sabrina R Nagel, Kumar S Raman, et al. Rayleigh–taylor instabilities in high-energy density settings on the national ignition facility. *Proceedings of the National Academy of Sciences*, 116(37):18233–18238, 2019.
9. VN Goncharov. Analytical model of nonlinear, single-mode, classical rayleigh-taylor instability at arbitrary atwood numbers. *Physical review letters*, 88(13):134502, 2002.
10. David L Youngs. Numerical simulation of turbulent mixing by rayleigh-taylor instability. *Physica D: Nonlinear Phenomena*, 12(1-3):32–44, 1984.
11. Brian K Spears, James Brase, Peer-Timo Bremer, Barry Chen, John Field, Jim Gaffney, Michael Kruse, Steve Langer, Katie Lewis, Ryan Nora, et al. Deep learning: A guide for practitioners in the physical sciences. *Physics of Plasmas*, 25(8):080901, 2018.
12. Kelli D Humbird, Jayson Luc Peterson, BK Spears, and RG McClarren. Transfer learning to model inertial confinement fusion experiments. *IEEE Transactions on Plasma Science*, 2019.
13. A Gonoskov, Erik Wallin, A Polovinkin, and I Meyerov. Employing machine learning for theory validation and identification of experimental conditions in laser-plasma physics. *Scientific reports*, 9(1):7043, 2019.
14. Gonzalo Avaria, Jorge Ardila-Rey, Sergio Davis, Luis Orellana, Benjamín Cevallos, Cristian Pavez, and Leopoldo Soto. Hard x-ray emission detection using deep learning analysis of the radiated uhf electromagnetic signal from a plasma focus discharge. *IEEE Access*, 7:74899–74908, 2019.

15. Kelli Denise Humbird. *Machine Learning Guided Discovery and Design for Inertial Confinement Fusion*. PhD thesis, 2019.
16. Jim A Gaffney, Scott T Brandon, Kelli D Humbird, Michael KG Kruse, Ryan C Nora, J Luc Peterson, and Brian K Spears. Making inertial confinement fusion models more predictive. *Physics of Plasmas*, 26(8):082704, 2019.
17. Bogdan Kustowski, Jim A Gaffney, Brian K Spears, Gemma J Anderson, Jayaraman J Thiagarajan, and Rushil Anirudh. Transfer learning as a tool for reducing simulation bias: Application to inertial confinement fusion. *IEEE Transactions on Plasma Science*, 2019.
18. Yeun Jung Kim, Minsoo Lee, and Hae June Lee. Machine learning analysis for the soliton formation in resonant nonlinear three-wave interactions. *Journal of the Korean Physical Society*, 75(11):909–916, 2019.
19. A Gonoskov. Employing machine learning in theoretical and experimental studies of high-intensity laser-plasma interactions.
20. Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.
21. Maziar Raissi, Zhicheng Wang, Michael S Triantafyllou, and George Em Karniadakis. Deep learning of vortex-induced vibrations. *Journal of Fluid Mechanics*, 861:119–137, 2019.
22. Arvind T Mohan and Datta V Gaitonde. A deep learning based approach to reduced order modeling for turbulent flow control using lstm neural networks. *arXiv preprint arXiv:1804.09269*, 2018.
23. Zheng Wang, Dunhui Xiao, Fangxin Fang, Rajesh Govindan, Christopher C Pain, and Yike Guo. Model identification of reduced order fluid dynamics systems using deep learning. *International Journal for Numerical Methods in Fluids*, 86(4):255–268, 2018.
24. Kjetil O Lye, Siddhartha Mishra, and Deep Ray. Deep learning observables in computational fluid dynamics. *Journal of Computational Physics*, page 109339, 2020.
25. J Nathan Kutz. Deep learning in fluid dynamics. *Journal of Fluid Mechanics*, 814:1–4, 2017.
26. Hengfeng Huang, Bowen Xiao, Huixin Xiong, Zeming Wu, Yadong Mu, and Huichao Song. Applications of deep learning to relativistic hydrodynamics. *Nuclear Physics A*, 982:927–930, 2019.
27. WC Wan, Guy Malamud, A Shimony, CA Di Stefano, MR Trantham, SR Klein, D Shvarts, CC Kuranz, and RP Drake. Observation of single-mode, kelly-helmholtz instability in a supersonic flow. *Physical review letters*, 115(14):145001, 2015.
28. B Fryxell, CC Kuranz, RP Drake, MJ Grosskopf, A Budde, T Plewa, N Hearn, JF Hansen, AR Miles, and J Knauer. The possible effects of magnetic fields on laser experiments of rayleigh–taylor instabilities. *High Energy Density Physics*, 6(2):162–165, 2010.
29. Carolyn C Kuranz, H-S Park, Channing M Huntington, Aaron R Miles, Bruce A Remington, T Plewa, MR Trantham, HF Robey, Dov Shvarts, A Shimony, et al. How high energy fluxes may affect rayleigh–taylor instability growth in young supernova remnants. *Nature communications*, 9(1):1–6, 2018.
30. CM Huntington, A Shimony, M Trantham, CC Kuranz, D Shvarts, CA Di Stefano, FW Doss, RP Drake, KA Flippo, DH Kalantar, et al. Ablative stabilization of

- rayleigh-taylor instabilities resulting from a laser-driven radiative shock. *Physics of Plasmas*, 25(5):052118, 2018.
31. RayleAI Database. <https://github.com/scientific-computing-nrcn/RayleAI>. [Online].
 32. Ygal Klein. Construction of a multidimensional parallel adaptive mesh refinement special relativistic hydrodynamics code for astrophysical applications. *Master's Thesis*, 2010.
 33. Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.
 34. Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, pages 2172–2180, 2016.
 35. Gan-Image-Similarity code repository. <https://github.com/marcbelmont/gan-image-similarity>. [Online].
 36. Savvas A. Chatzichristofis Mathias Lux. Lire: Lucene image retrieval - an extensible java cbir library.
 37. Dengsheng Zhang, Aylwin Wong, Maria Indrawan, and Guojun Lu. Content-based image retrieval using gabor texture features. *IEEE Transactions Pami*, 13, 2000.
 38. Dr. Antony Selvadoss Thanamani K. Haridas. Well-organized content based image retrieval system in rgb color histogram, tamura texture and gabor feature.
 39. Savvas A Chatzichristofis and Yiannis S Boutalis. Fcth: Fuzzy color and texture histogram-a low level feature for accurate image retrieval. In *2008 Ninth International Workshop on Image Analysis for Multimedia Interactive Services*, pages 191–196. IEEE, 2008.
 40. Stéphane Lathuilière, Pablo Mesejo, Xavier Alameda-Pineda, and Radu Horaud. A comprehensive analysis of deep regression. *IEEE transactions on pattern analysis and machine intelligence*, 2019.
 41. Philipp Fischer, Alexey Dosovitskiy, and Thomas Brox. Image orientation estimation with convolutional networks. In *German Conference on Pattern Recognition*, pages 368–378. Springer, 2015.
 42. Siddharth Mahendran, Haider Ali, and René Vidal. 3d pose regression using convolutional neural networks. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 2174–2182, 2017.
 43. Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, pages 265–283, 2016.
 44. Roberto Brunelli. *Template matching techniques in computer vision: theory and practice*. John Wiley & Sons, 2009.
 45. Wael Abd-Almageed Premkumar Natarajan Jiaxin Cheng, Yue Wu. Qatm: Quality-aware template matching for deep learning. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019.
 46. SHI Xingjian, Zhouong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Advances in neural information processing systems*, pages 802–810, 2015.

47. Yunbo Wang, Mingsheng Long, Jianmin Wang, Zhifeng Gao, and S Yu Philip. Predrnn: Recurrent neural networks for predictive learning using spatiotemporal lstms. In *Advances in Neural Information Processing Systems*, pages 879–888, 2017.
48. Chris Ding and Xiaofeng He. K-means clustering via principal component analysis. In *Proceedings of the twenty-first international conference on Machine learning*, page 29, 2004.
49. Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.
50. WC Wan, Guy Malamud, A Shimony, CA Di Stefano, MR Trantham, SR Klein, D Shvarts, RP Drake, and CC Kuranz. Observation of dual-mode, kelvin-helmholtz instability vortex merger in a compressible flow. *Physics of Plasmas*, 24(5):055705, 2017.
51. A Shimony, WC Wan, SR Klein, CC Kuranz, RP Drake, D Shvarts, and G Malamud. Construction and validation of a statistical model for the nonlinear kelvin-helmholtz instability under compressible, multimode conditions. *Physics of Plasmas*, 25(12):122112, 2018.
52. Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. A survey on deep transfer learning. In *International conference on artificial neural networks*, pages 270–279. Springer, 2018.
53. NegevHPC Project. <http://www.negevnhpc.com>. [Online].