Self-Supervised Learning Strategies for a Platform to Test the Toxicity of New Chemicals and Materials

Thomas Lautenschlager[1], Nils Friederich[1,2], Angelo Jovin Yamachui Sitcheu[1], Katja Nau[1], Gaëlle Hayot[2], Thomas Dickmeis[2], Ralf Mikut[1]

[1]Institute for Automation and Applied Informatics (IAI)
[2]Institute for Biological and Chemical Systems - Biological Information Processing (IBCS-BIP)
Karlsruhe Institute of Technology
E-Mail: thomas.lautenschlager@kit.edu

Abstract

High-throughput toxicity testing offers a fast and cost-effective way to test large amounts of compounds. A key component for such systems is the automated evaluation via machine learning models. In this paper, we address critical challenges in this domain and demonstrate how representations learned via self-supervised learning can effectively identify toxicant-induced changes. We provide a proof-of-concept that utilizes the publicly available EmbryoNet dataset, which contains ten zebrafish embryo phenotypes elicited by various chemical compounds targeting different processes in early embryonic development. Our analysis shows that the learned representations using self-supervised learning are suitable for effectively distinguishing between the modes-of-action of different compounds. Finally, we discuss the integration of machine learning models in a physical toxicity testing device in the context of the TOXBOX project.

1 Introduction

The REACH (Registration, Evaluation, Authorisation and Restriction of Chemicals) regulation, introduced in 2007, aims to better understand chemical compounds entering the EU (European Union) market [1]. According to REACH, companies that import or produce a certain compound in quantities exceeding one tonne, are obligated to test the compounds for toxicity and report the results to the European Chemicals Agency (ECHA) [1]. Over 23 000 different compounds were registered under REACH as of 2025 [2].
This illustrates the need for large amounts of toxicity tests to be conducted. Typically, these tests are done *in vivo* using rats or other mammals [3]. However, they are relatively costly due to housing and feeding needed, as well as the comparably low reproduction rate of these animals. Furthermore, animal testing requires lengthy legal procedures and poses ethical concerns. Russell and Burch formulated the 3R principles that aim to replace, reduce and refine tests conducted on animals, where possible [4]. Consequently, interest in alternative forms of toxicity testing is increasing [3].
For approaches deviating from traditional *in vivo* studies, the umbrella term of New Approach Methodologies (NAMs) was coined. In the context of NAMs, tests that are suitable for High-Throughput Screening (HTS) are often discussed. These include for example *in vivo* studies using zebrafish (*Danio rerio*) embryos [5]. They are cheaper to rear due to lower maintenance costs and a higher progeny number than animals traditionally used in toxicity testing. Furthermore, according to EU legislation, zebrafish embryos are not considered animals up until 5 days post fertilization (DPF), facilitating easier adoption for testing [6]. *Daphnia magna* is another species that is often considered for NAMs and HTS [7]. Apart from *in vivo* testing, *in vitro* tests using cell-based assays or organ models are also rising in popularity [3]. While cell-based assays are suitable for HTS experiments, the viability of organ models for HTS is being actively investigated [8]. More complex approaches, such as organs-on-a-chip or extensions using multiple connected organs for body-on-a-chip systems, are discussed as well [9].
The vast amount of data generated by such HTS approaches, however, necessitates the use of automated evaluation methods, which can be achieved using ML models. While most of the literature on ML in toxicology is focused

on *in silico* models, research on ML for toxicity test automation is fairly scarce [10]. However, since *in silico* ML models often show poor generalization to compounds with dissimilar properties to the ones they were trained on [10], there is a need for automatic evaluations of experimental HTS data, using ML. The data generated through HTS is often high-dimensional, encompassing microscopic images and time-series data such as electrochemical readouts. Since Deep Learning (DL) models generally perform better on high-dimensional data than traditional ML models [11], they are better suited for evaluating HTS data. Incorporating Self-Supervised Learning (SSL) into DL models can offer various advantages in the domain of toxicity testing. Since labeled data is often scarce, self-supervised pretraining is a valuable technique for building robust models using smaller datasets for downstream tasks such as classification. Furthermore, the continuous representations learned by SSL can model concentration-dependent gradients of toxicant-induced changes, while the inherent clustering of the learned representations can identify compounds with similar modes-of-action.

Recently, the use of Large Language Models (LLMs) has also been discussed in the context of toxicity predictions [12]. LLMs are mainly considered in the context of data extraction and data curation from different toxicological databases or from scientific literature [13]. They could also be utilized to directly make predictions based on a given literature database using Retrieval-Augmented Generation (RAG) or fine-tuned LLMs [14]. However, since LLMs are prone to hallucinations [15], their application and wider adoption should be done cautiously. Fusion approaches for ML models, combining different inputs, can also be explored. In this way, different experimental data as well as physicochemical properties of the tested compounds could be combined for a single toxicity prediction. A summary of the discussed approaches to computational models for toxicity predictions is given in Figure 1.

While the discussion of individual toxicological endpoints is beyond the scope of this paper, it is important to note that computational models often only make predictions on one or a few toxicological endpoints. ML models can only accurately predict toxicity for the endpoints for which the model was trained on. In this paper, we mostly focus on discussing image-based approaches because most of the published work on toxicity tests that are viable for HTS focuses on the automatic evaluation of images. The outline for the rest of this paper is

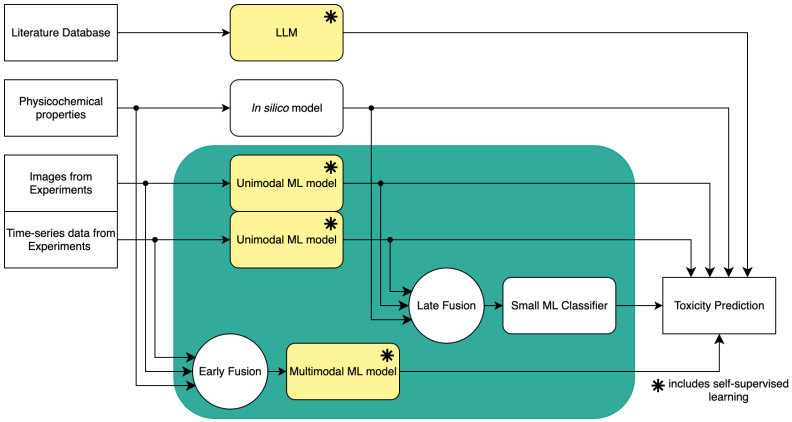

Figure 1: Diagram of different computational models for toxicity predictions. The green box denotes the approaches more closely discussed in this paper. Models that incorporate SSL are marked with an asterisk and highlighted in yellow.

as follows: In Section 2, we discuss existing DL approaches that are suitable for HTS scenarios. We provide a brief introduction to SSL in general and portray advantages of SSL specific to toxicity testing in Section 3. Section 4 contains a proof-of-concept demonstrating that representations learned by SSL can successfully identify toxicant-induced changes and relate the same modes-of-action to each other. Using various analyses, we investigate properties of the learned representations in detail and discuss them. Lastly, in Section 5, we address challenges arising from the integration of DL models into TOXBOX, a real-world toxicity testing device and outline strategies to tackle them.

2 Related Work

There are several works concerned with using DL models for evaluating toxicity tests, which we are going to discuss here. These works could be adapted for use in HTS scenarios. However, regarding species commonly used in toxicity testing, there is a notable lack of research that consistently focuses on a single aspect of toxicity testing for a specific species. Even less work focuses on the automation of established toxicity tests.

As already mentioned, zebrafish are a model organism often discussed in the context of toxicity testing [5]. Several test protocols have been established for zebrafish embryos and larvae <5 DPF. For example, the Fish Embryo Acute Toxicity (FET) Test is OECD-approved [16] and is often discussed as an alternative to the Fish Acute Toxicity Test [17], which uses adult fishes and is therefore not suitable for HTS. Various behavioral tests have also been established. These tests, though often lacking standardized protocols, enable the evaluation of neurotoxicity [18].

DL techniques for the classification of abnormally developing zebrafish embryos have been proposed in several studies [19–22]. However, the different investigations show low consistency. Most of these focus on the classification of hatched eleutheroembryos during different timepoints [19–21], only one publication focuses on earlier embryonic stages [22]. All the publications use different classes in their classification approaches. These inconsistencies in the existing research make comparisons difficult.

There are also no publications that tackle the automatic evaluation of the FET, the only OECD-approved toxicity test involving zebrafish <5 DPF that focuses on phenotypical changes [16]. However, the automatic identification of coagulated zebrafish embryos, one of the endpoints of the FET, has been addressed in several works [22, 23]. Existing approaches have also paid little attention to identifying modes-of-action of the compounds. Only Čapek et al. [22] define the classified phenotypes based on different developmental pathways that can be blocked by certain toxicants. These phenotypes, however, also do not cover all possible toxicant-induced changes.

Several papers focus on DL approaches for toxicity testing using *Daphnia magna*. They include models that determine and quantify morphological changes in *Daphnia magna* due to toxicant exposure [24], models to determine the size and growth rate [25] and approaches for tracking *Daphnia magna* [26] as well as identifying compounds based on locomotor tracks [27].

Few approaches exist for the use of DL models in cell assays or organ models for toxicity testing. One approach automatically detects the nuclei of the cells and classifies them as either 'healthy' or 'toxicity-affected' [28]. Another approach uses time-series data based on a cell impedance signal for the classification of different modes-of-action [29]. Cell tracking approaches such as [30, 31] are also suitable for toxicity testing, since features such as the number of cells or

size of cells can also be used to make predictions on the toxicity of a certain compound.
Hu et al. [32] use DL models for predicting the thickness of a skin model. They show that lower thickness of the epidermal layer can be used to predict skin toxicity.
To the best of our knowledge, only two studies are using SSL that can be considered for the automatic evaluation of toxicity tests. Toulany et al. [33] use a Twin Network trained with a triplet loss to investigate the embryonic development in zebrafish. The trained network can be used to determine the similarity between embryo images. This is used for identifying different developmental stages, comparisons regarding the development of zebrafish embryos under different temperatures and detecting deviations from normal development. The authors show that the model can also identify deviations from normal development that are toxicant-induced [33].
In the second paper on SSL in toxicity testing, Gendelev et al. [34] use Twin Networks on the Motion Index, a measurement of movement based on pixel intensity changes between frames in videos, from different behavioral tests using 7 dpf zebrafish larvae. This approach can group similar modes-of-action together.

3 Potential of Self-Supervised Learning in Toxicology

3.1 Overview of Self-Supervised Learning

SSL encompasses methods that use a pretext task for learning useful lower-dimensional representations [35]. The pretext task focuses on optimizing for a target t_{SSL} that can be generated from the data itself [36]. Depending on the self-supervised algorithm, the pretext tasks in computer vision can range from predicting the correct order of shuffled image patches [37], mapping two differently augmented views of the same image together [38], or reconstructing image patches that were masked in the input [39].
Typically, SSL uses an encoder that maps the inputs x to a latent space h. Often,

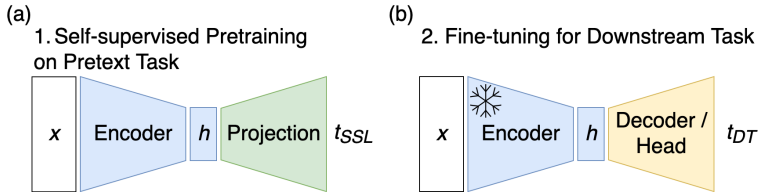

Figure 2: Simplified diagram of (a) pretraining via SSL and (b) fine-tuning for downstream task. The encoder is usually frozen during fine-tuning. x denotes the input of the model, h the learned latent space, and t_{SSL} and t_{DT} the targets of the self-supervised learning task and the downstream task, respectively.

some kind of projection is used on the latent space h, the output of which is used for the optimization regarding the target t_{SSL}. This can, for example, be a linear layer [40], a projection head [38] or a decoder [39].

The desired output of the SSL models are the lower-dimensional representations of the data in the learned latent space h. Generally, in the latent space h, representations from similar inputs are mapped closely together, while representations from dissimilar inputs are mapped further away from each other. In computer vision, these representations are often referred to as visual representations. For the sake of brevity and because SSL can also be used to attain lower-dimensional representations from non-image data, we use the term 'representations' for the rest of this paper.

In general computer vision tasks, SSL is often used for pretraining [35]. The learned representations are then used to fine-tune for a specific downstream task, such as classification or segmentation [35]. Depending on the kind of task, a decoder or head is used on top of the usually frozen encoder. The decoder or head is then trained regarding the target t_{DT} of the downstream task. This typical training procedure is pictured in Figure 2. Popular SSL methods for images include: SimCLR [38], MoCo and its extensions [40–42], BYOL [43], SwAV [44], DINO and its extensions [45–47], Masked Autoencoders [39] and SimSiam [48].

3.2 Label-Efficient Model Training

Using pretrained models for training on downstream tasks can also be applied to toxicity testing. This allows for the training of more label-efficient models. Further, the representations learned by SSL are often more generalizable, enabling transfer learning, where the representations from a different, but similar dataset can be used for fine-tuning to a task, where both data and labels are scarce.
Since few image datasets for toxicity testing are published and generating new datasets is expensive, leveraging similar datasets in SSL pretraining makes research on ML models for toxicity testing more feasible. A strategy that could be used to identify suitable datasets was outlined by Yamachui Sitcheu et al. [49].

3.3 Continuous Representations

The latent space h learned by SSL methods offers several additional advantages for toxicity testing. A problem when applying supervised DL models to toxicity tests is that the methods often used do not account for properties specific to toxicity testing.
For example, classification often falls short when evaluating toxicant-induced morphological changes. These changes are usually continuous and the cutoff point is often based on observer experience, with little to no standardization. Thresholds of toxicant-induced changes can therefore vary between studies. It is often unclear whether small changes are already labeled as 'toxic' or only if a clear abnormal phenotype can be observed. Additionally, there are studies that use several classes [22, 24] for different magnitudes of the same phenotypic changes, resulting in even more hazily defined cutoff points. By mapping the samples into a latent space that allows for continuous representations of the morphological changes, SSL offers an elegant solution for this problem. The resulting representation not only allows fine differentiation based on phenotypic changes but can be thought of as concentration-dependent gradients. The learned representations of a certain phenotypical change will be mapped into the same direction away from healthy phenotypes. Since toxicant-induced changes get

more severe with higher concentrations of the compound, the distance to the healthy phenotypes will increase with higher concentrations of the compound.
Another downside of supervised classification is that biases can be introduced if small changes due to a toxicant are not represented in the labels used for training. For example, the EmbryoNet-Prime DL model, trained on data with labels shifted 4 hours into the past, can identify morphological changes earlier than the expert who labeled the data [22], indicating that small phenotypical changes are already present.
However, simply shifting the labels can result in false labels, since it is unclear when the phenotypical changes first occur. This can deteriorate model performance. SSL could potentially identify when small phenotypical changes occur without label-induced biases.

3.4 Identification of Similar Modes-of-Action

The clustering inherent to SSL can also be used for the identification of similar modes-of-action. Since similar images will be mapped together and dissimilar images mapped away from each other, similar phenotypes induced by compounds with a similar mode-of-action will be clustered. Gendelev et al. [34] have shown that this is possible using time-series data of pixel intensity changes from behavioral zebrafish tests.
Additionally, a classifier without rejection class forces unknown toxicant-induced changes into one of the classes known from the training dataset. In the case of SSL, the representations of an unknown morphological change are mapped away from the representations of the known classes, making it apparent that the representations do not belong to any of the known classes.

4 Preliminary Experiment: Proof-of-Concept

4.1 Methods

We chose SimCLR [38] for our proof-of-concept investigating the properties of the latent space of a SSL model trained on image data showing toxicant-induced

morphological changes. SimCLR is an important baseline in the field of SSL and learns meaningful representations of the data. The self-supervised target t_{SSL} of SimCLR aims at minimizing the distance between the representations of two differently augmented views of the same image while maximizing the distance between views of different images. SimCLR uses cosine similarity as the distance measure between different views [38]. The resulting latent space is a hypersphere on which the representations are mapped.

We used ResNet50 [50] as the backbone, which we trained using SimCLR [38]. After the training, we evaluated the latent space using linear probing, where a linear classifier is trained on the representations that the frozen backbone outputs. This method is a standard procedure in SSL research to assess the quality of learned representations.

Since SimCLR training is unsupervised, the labels of the dataset are only used for training the linear classifier. This usually results in worse performance than training the network fully supervised. However, it can still be useful to evaluate the success of self-supervised training.

To better understand the latent space and the representations SimCLR learned, we visualize the latent space using UMAP for dimensionality reduction [52].

Furthermore, we investigate the representations of each class. Through a forward pass using the training dataset, we obtain the representations for the training dataset. We calculate the centers of each class by taking the mean of the respective class representations

$$\boldsymbol{c}_k = \frac{1}{|C_k|} \sum_{i \in C_k} \boldsymbol{h}_i \tag{1}$$

where $\boldsymbol{h}_i$ denotes one learned representation, $\boldsymbol{c}_k$ represents the center $\boldsymbol{c}$ for class k, C_k denotes the set of representations belonging to class k and $|C_k|$ its cardinality. The dimensionality of $\boldsymbol{h}$ and $\boldsymbol{c}$ are dependent on the type of network used. Since we use ResNet50 as a backbone, the resulting dimensionality for $\boldsymbol{h}$ and $\boldsymbol{c}$ is 2048 in our analyses.

After the calculation, the centers are normalized

$$\tilde{\boldsymbol{c}}_k = \frac{\boldsymbol{c}_k}{\|\boldsymbol{c}_k\|} \tag{2}$$

where $\tilde{\boldsymbol{c}}_k$ refers to the normalized class center of class k and $\|\boldsymbol{c}_k\|$ to the Euclidean norm of class center $\boldsymbol{c}_k$.
Next, the class centers are used to calculate the mean cosine similarity of the representations of each class to each center

$$\overline{\text{sim}_{\cos}}(C_l;\tilde{\boldsymbol{c}}_k) = \frac{1}{|C_l|}\sum_{i\in C_l} \boldsymbol{h}_i^\top \cdot \tilde{\boldsymbol{c}}_k \tag{3}$$

where $\overline{\text{sim}_{\cos}}(C_l;\tilde{\boldsymbol{c}}_k)$ denotes the mean cosine similarity between the set C_l that includes the representations $\boldsymbol{h}_i$ of the test dataset. Note that the representations of $\boldsymbol{h}_i$ and $\tilde{\boldsymbol{c}}_k$ are both normalized in this calculation.
The distance of a point to a class center can be thought of as an anomaly score for that class. To achieve a deeper understanding of the constructed latent space, we also calculate the cosine similarities between the different class centers:

$$\text{sim}_{cos}(\tilde{\boldsymbol{c}}_l;\tilde{\boldsymbol{c}}_k) = \tilde{\boldsymbol{c}}_l^\top \cdot \tilde{\boldsymbol{c}}_k \tag{4}$$

4.2 Dataset

For our analyses, we used the publicly available EmbryoNet dataset [22]. It features images of ten different zebrafish embryo phenotypes. There are seven phenotypes, where the used toxicant targeted a major signaling pathway in early embryonic development. The respective phenotypes are named after their affected pathway and whether a loss-of-function or gain-of-function is present: -BMP, +RA, -Wnt, -FGF, -Nodal, -Shh and -PCP. Other classes include the 'Normal' class, featuring normally developing embryos, the 'Dead' class, which includes embryos that have died and coagulated, and the 'Unknown' class, for embryos whose phenotype could not be identified.
The original dataset features embryos that are periodically imaged from 2 hours post fertilization (HPF) to 26 HPF, since developmental aspects are closely discussed in the EmbryoNet paper [22]. However, since we were most interested in classifying the different phenotypes that only become apparent as development progresses, we chose to use only images from the later timepoints. This not only reduced training time but also avoided learning visual features that are not necessary for phenotype classification.

The EmbryoNet dataset consists of images of wells containing multiple zebrafish embryos [22]. We adopted the predefined split between training and test dataset of the EmbryoNet dataset. Additionally, we defined a validation dataset using images from 10% of the wells that make up the training dataset. The chosen wells were randomly sampled. For extracting the individual embryo crops from the well images, we used the bounding boxes, which are provided with the dataset. For the training and validation dataset we used the crops of embryos ranging from 25 HPF to 26 HPF. The evaluations were done using only the last crop of each embryo in the test dataset, which was recorded at 26 HPF. The resulting training, validation and test datasets consist of 135 475, 15 670 and 772 embryo crops, respectively.

4.3 Implementation Details

We used MMPretrain [53] for training SimCLR and the linear classifier. The training was done using 8 NVIDIA A100-40s. Other evaluations were done using custom code and were run on an NVIDIA RTX 3090. The code is available at `github.com/lautthom/self_supervised_learning_strategies_toxicity_testing`. Details regarding the hyperparameters and augmentations used are available in the config files used for MMPretrain provided with the code.

Unless otherwise noted, we used the same hyperparameters as described in the original SimCLR paper [38]. We trained SimCLR for 200 epochs and reduced the batch size to 2048. The learning rate was adjusted accordingly with square root scaling [51].

For the linear classifier, we also adopted the training and testing procedure as well as the hyperparameters as defined in the SimCLR paper [38], unless otherwise noted. Since the classes in the EmbryoNet dataset are imbalanced, we used a weighted loss function for the training of the linear classifier

$$w_i = \frac{n}{n_i \cdot C} \tag{5}$$

where w_i specifies the weight of class i in the loss function, n the number of total samples, n_i the number of samples in class i and C the number of classes.

Furthermore, we used early stopping based on the accuracy the linear classifier achieved on the validation dataset.
The augmentations were adjusted to fit the domain-specific needs of zebrafish embryo images. The following augmentations were used for both the SimCLR training and the linear classifier training: random crop, horizontal flip, rotations of up to 360°, random brightness changes, random contrast changes, CLAHE, sharpen, motion blur, defocus, grid distortion, optical distortion, elastic transform, salt and pepper noise, Gaussian noise, Poisson noise and solarize.

4.4 Results

The linear classifier trained on top of the representations learned by SimCLR achieved an accuracy of 79.9% on the test dataset. This is about ten percentage points below the 89% accuracy reported in the original EmbryoNet paper [22]. The normalized confusion matrix of the linear classifier trained on the 10 classes is depicted in Figure 3. Dead embryos are classified most reliably with a recall of 100%. Other classes with a high recall are the -BMP, -FGF, -Nodal, +RA and -Wnt phenotypes. The 'Normal' class has a relatively low recall of 60%. The -PCP and -Shh phenotypes also have a low recall. The 'Unknown' phenotype has the lowest recall, however, only 4 images belong to the 'Unknown' class in our test dataset.

An UMAP visualization of the learned representations of the test dataset is given in Figure 4. The visualization reflects the recall values given in Figure 3. The 'Dead' class is mapped far away from the other classes. The representations of the other classes with a high recall are also mapped close to each other and are fairly easy to distinguish from the representations of other classes.

For each class, we calculated the mean cosine similarity between the representations and the centers of the respective class. The results are given in Figure 5. The lowest mean cosine similarity is 0.70. However, for all classes, the mean cosine similarities between their representations and their respective class centers are higher than the mean cosine similarities to all other class centers.

Figure 6 shows the cosine similarities between the different class centers. As in Figure 5 all cosine similarities are fairly high. The lowest cosine similarity between two class centers is 0.86.

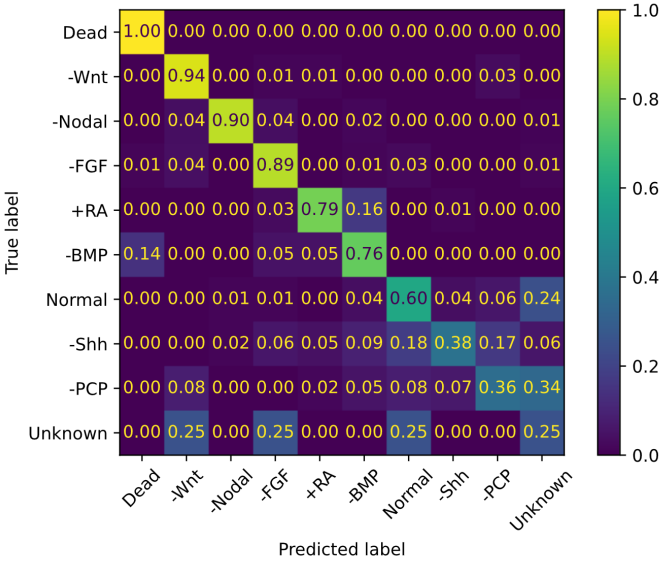

Figure 3: Confusion matrix of the linear classifier trained using SimCLR representations

Due to the high cosine similarities between the centers and representations in Figure 5 as well as between the centers themselves in Figure 6, we took a closer look at the cosine similarities of the individual representations. The minimal cosine similarity between two representations is 0.37, while the mean similarity is 0.64 and the highest cosine similarity is 1.0. This means that only a small part of the hypersphere that the images are mapped to during SSL training is populated by the representations.

4.5 Discussion, Limitations and Outlook

Our investigation of the learned latent space in Section 4.4 leads to different insights. The performance of the linear classifier shows that the learned representations have a fairly good quality. While the classification is markedly worse than the supervised baseline, it still reaches acceptable performance. Since most of the classes correspond to a certain mode-of-action, the clustering

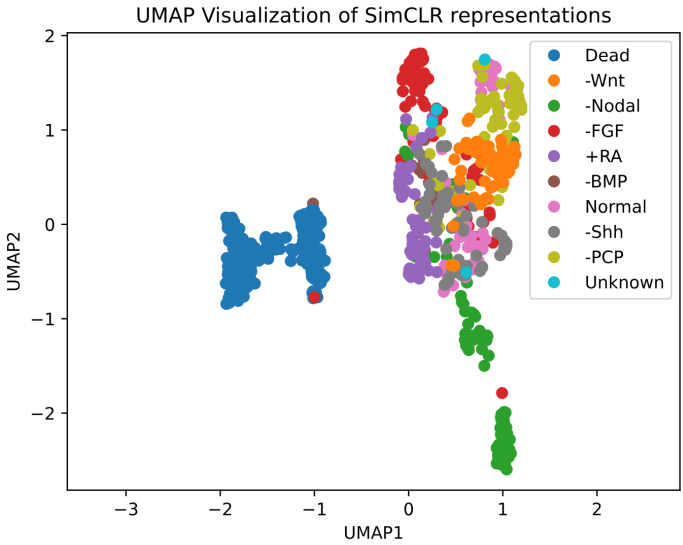

Figure 4: UMAP visualization of SimCLR representations

based on modes-of-action works fairly well. However, the phenotypes in the EmbryoNet dataset are elicited by the same compounds, which makes it hard to evaluate if the clustering was indeed based on the mode-of-action or on some other compound-specific properties. Certain phenotypes reach a particular low recall, for example the -Shh and -PCP phenotype. However, this is also true for the supervised model, as well as for the evaluations of experienced developmental biologists, reported in the EmbryoNet paper [22].

An interesting approach for future research could be to evaluate the representations learned via SSL on other downstream tasks, such as segmentation or transfer learning to another classification task. Presumably, the representations learned by SSL should outperform the ones learned during supervised training in these tasks. A possible application would be the automatic evaluation of the FET [16]. Future investigations could also explore fine-tuning the models with fewer available labels and compare the performance deterioration to fully supervised learning.

A problem with the present latent space is that the representations only populate

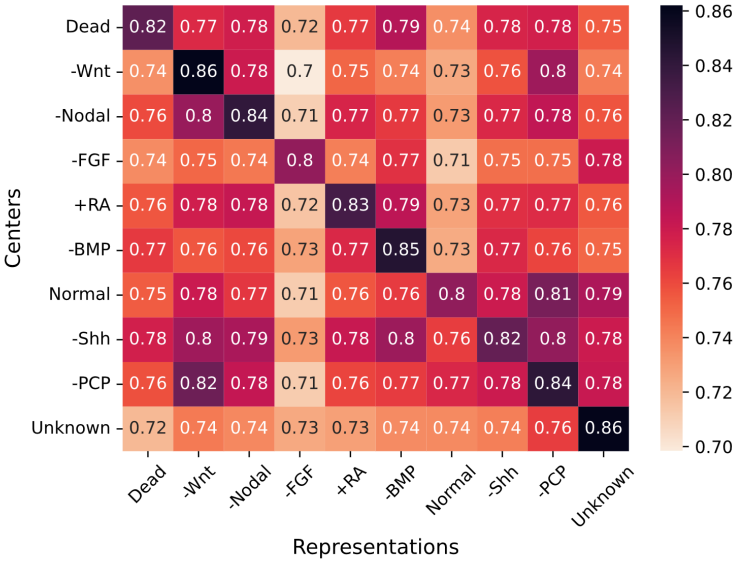

Figure 5: Mean cosine similarities between class centers and representations of the class

a small area of the hypersphere. Investigations show that more uniform distributions on the hypersphere generally improve performance [54]. It is unclear whether this is also true for domain-specific use cases, where the images are very similar to each other.

Further investigations are needed to see if the theoretical considerations presented in Section 3 can be empirically verified. Clustering based on similar modes-of-action was already shown to be feasible for one other application [34] and the results presented in this paper support this finding. Given this, it is likely that concentration-dependent gradients in the latent space also exist. Unfortunately, the EmbryoNet dataset is not best suited for this investigation, since the authors used the same concentrations for the elicitation of most of the phenotypes [22].

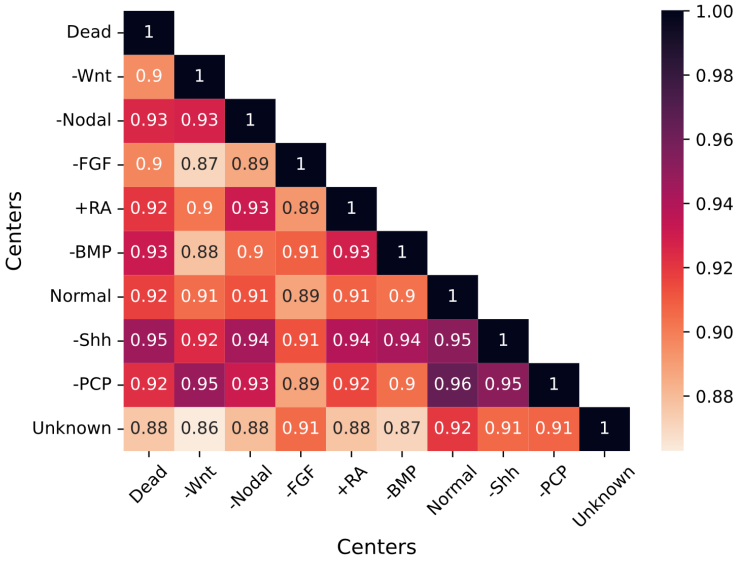

Figure 6: Cosine similarities between the class centers

5 Integration with TOXBOX device

Since the aim is to integrate ML models in high-throughput processes, the ML models also need to be integrated with a physical toxicity testing device. We discuss this in the context of the TOXBOX[1] project. The project aims to design an all-in-one platform for reliable toxicity testing [55]. A prototype of the TOXBOX device is pictured in Figure 7.

TOXBOX will feature different *in vitro* organ models as well as a zebrafish embryo module [55]. Due to the advantages illustrated in this paper, pretraining the models via SSL and then fine-tuning them to the specific toxicity prediction task seems to be the most viable option. This should be especially advantageous if the data and/or labels generated during the TOXBOX project are scarce and data from similar datasets can be leveraged using SSL. Fully supervised models should be trained as well and compared to SSL models, to ensure that the model

[1] https://toxbox.eu

Figure 7: Image of the TOXBOX prototype; screenshot from [56]

with the best performance will be used.

Explainable Artificial Intelligence (XAI) methods can help in the evaluation of the different ML models [57]. Since XAI makes the underlying factors that lead to a certain prediction of a ML model more transparent, experts can assess whether the factors used are indicative of toxicity.

Furthermore, the latent space of the SSL models could be used to gain more information about the tested compound. Based on the distance between the representations of known compounds and the tested compound, it can be determined whether the compound has a similar mode-of-action as known compounds or whether it has an unknown effect. Compounds with unknown effects merit more thorough investigation, both on the compounds themselves and also if something went wrong during testing.

A crucial topic to discuss when using ML models in toxicity testing is concept drift. Concept drift refers to gradual changes in the underlying data, which happen over time and can deteriorate the model's performance [58]. This can happen especially easily in toxicity testing if groups of compounds are tested that differ in important aspects from those used in acquiring the training data, particularly when compounds with different modes-of-action are tested. This means that the model's performance needs to be closely monitored to ensure reliable predictions.

If new modes-of-action are found or the model's performance drops due to concept drift, it may be necessary to retrain the model. This can be challenging, as the newly acquired data from the device may be highly imbalanced. Further, the data of the previously unknown mode-of-action can be scarce. Different strategies for retraining the models should be explored and closely evaluated in such scenarios.

6 Conclusion

In this paper, we have illustrated how aspects inherent to SSL are suitable to address different challenges specific to toxicity testing, specifically dealing with sparse labeled data, accounting for continuous changes due to toxicant exposure and identifying similar modes-of-action. We provided a proof-of-concept that demonstrates how representations learned via SSL can, in practice, be utilized for toxicity testing. Further, we discussed various challenges involved in adapting machine learning models to physical toxicity testing devices.

7 Acknowledgments

This work has received funding from the European Union's HORIZON-CL4-2023-RESILIENCE-01 Research and Innovation programme under Grant Agreement No 101138387 (TOXBOX). Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union. The European Union cannot be held responsible for them.
This work is supported by the Helmholtz Association Initiative and Networking Fund on the HAICORE@KIT partition.
The present contribution is supported by the Helmholtz Association under the joint research school "HIDSS4Health - Helmholtz Information and Data Science School for Health." and the program "Natural, Artificial and Cognitive Information Processing (NACIP)".

7.1 Author contributions

The authors used the AI language models GPT-4.5 and GPT-5.0 to improve the language and style of this manuscript. The authors have accepted responsibility for the entire content of this manuscript and approved its submission. We describe the individual contributions of T. Lautenschlager (TL), N. Friederich (NF), A.J. Yamachui Sitcheu (AJYS), K. Nau (KN), G. Hayot (GH), T. Dickmeis (TD), R. Mikut (RM): Conceptualization: TL, NF, RM; Methodology: TL, NF, RM; Software: TL; Investigation: TL; Writing - Original Draft: TL; Writing - Review & Editing: TL, NF, AJYS, KN, GH, TD, RM; Supervision: NF, RM; Project administration: RM; Funding Acquisition: TD, RM.

References

[1] "REACH Regulation," European Commission, [Online]. Available: https://environment.ec.europa.eu/topics/chemicals/reach-regulation_en. [Accessed: Jul. 18, 2025].

[2] "REACH Registration statistics," European Chemicals Agency (ECHA), [Online]. Available: https://echa.europa.eu/documents/10162/2741157/registration_statistics_en.pdf/58c2d7bd-2173-4cb9-eb3b-a6bc14a6754b. [Accessed: Jul. 18, 2025].

[3] "The use of alternatives to testing on animal for the REACH Regulation," European Chemicals Agency (ECHA), [Online]. Available: https://op.europa.eu/en/publication-detail/-/publication/2203ad8b-1ec9-11ee-806b-01aa75ed71a1/. [Accessed: Aug. 22, 2025]

[4] W. M. S. Russell and R. L. Burch. "The Principles of Humane Experimental Technique". London: Methuen, 1959.

[5] L. Yang, N. Y. Ho, R. Alshut, J. Legradi, C. Weiss, M. Reischl, R. Mikut, U. Liebel, F. Müller, and U. Strähle. Zebrafish embryos as models for embryotoxic and teratological effects of chemicals. *Reproductive Toxicology*, vol. 28, no. 2, pp. 245–253, Sep. 2009.

[6] European Union. Directive 2010/63/EU of the European Parliament and of the Council of 22 September 2010 on the protection of animals used for scientific purposes. *Official Journal of the European Union*, vol. L 276, pp. 33–79, Sep. 2010.

[7] A. Tkaczyk, A. Bownik, J. Dudka, K. Kowal, and B. Ślaska. *Daphnia magna* model in the toxicity assessment of pharmaceuticals: A review. *Science of The Total Environment*, vol. 763, Apr. 2021, Art no. 143038.

[8] A. Astashkina, B. Mann, and D. W. Grainger. A critical evaluation of in vitro cell culture models for high-throughput drug screening and toxicity. *Pharmacology & Therapeutics*, vol. 134, no. 1, pp. 82–106, Apr. 2012.

[9] S. Yang, Z. Chen, Y. Cheng, T. Liu, L. Yin, Y. Pu, and G. Liang. Environmental toxicology wars: Organ-on-a-chip for assessing the toxicity of environmental pollutants. *Environmental Pollution*, vol. 268, no. B, Jan. 2021, Art no. 115861.

[10] A. H. Vo, T. R. Van Vleet, R. R. Gupta, M. J. Liguore, and M. S. Rao. An Overview of Machine Learning and Big Data from Drug Toxicity Evaluation. *Chemical Research in Toxicology*, vol. 33, no. 1, pp. 20–37, Oct. 2019.

[11] I. Goodfellow, Y. Bengio, and A. Courville. "Deep Learning". Cambridge, MA: MIT Press, 2017.

[12] M. Corradi, T. Luechtefeld, A. M. de Haan, R. Pieters, J. H. Freedman, T. Vanhaecke, M. Vinken, and M. Teunis. The application of natural language processing for the extraction of mechanistic information in toxicology. *Frontiers in Toxicology*, vol. 6, May 2024, Art no. 1393662.

[13] A. Sonnenburg, B. van der Lugt, J. Rehn, P. Wittkowski, K. Bech, F. Padberg, D. Eleftheriadou, T. Dobrikov, H. Bouwmeester, C. Mereu, F. Graf, C. Kneuer, N. I. Kramer, and T. Blümmel. Artificial intelligence-based data extraction for next generation risk assessment: Is fine-tuning of a large language model worth the effort? *Toxicology*, vol. 508, Nov. 2024, Art no. 153933.

[14] A. R. Kattamreddy, and H. Chinnam. The future of large language models in toxicological risk assessment: Opportunities and challenges. *Public Health and Toxicology*, vol. 5, pp. 1–3, Jan. 2025.

[15] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, and P. Fung. Survey of Hallucination in Natural Language Generation. *ACM Computing Surveys*, vol. 55, no. 12, pp. 1–38, Mar. 2023.

[16] OECD. "Test No. 236: Fish Embryo Acute Toxicity (FET) Test," in "OECD Guidelines for the Testing of Chemicals, Section 2," OECD Publishing, Paris, 2025.

[17] OECD. "Test No. 203: Fish, Acute Toxicity Test," in "OECD Guidelines for the Testing of Chemicals, Section 2," OECD Publishing, Paris, 2025.

[18] J. Legradi, N. el Abdellaoui, M. van Pomeren, and J. Legler. Comparability of behavioural assays using zebrafish larvae to assess neurotoxicity. *Environmental Science and Pollution Research*, vol. 22, pp. 16277–16289, 2015.

[19] B. Wang, Q. Sun, Y. Liu, J. Zhang, G. Li, S. Wu, H. Zheng, J. Ye, M. Zhou, H. Zheng, Y. Yu, Y. Zhong, Y. Wu, D. Huang, B. Wang, and Z. Weng. Intelligent larval zebrafish phenotype recognition via attention mechanism for high-throughput screening. *Computers in Biology and Medicine*, vol. 188, Apr. 2025, Art no. 109892.

[20] S. Shang, S. Lin, and F. Cong. Zebrafish Larvae Phenotype Classification from Bright-field Microscopic Images Using a Two-Tier Deep-Learning Pipeline. *Applied Sciences*, vol. 10, no. 4, Feb. 2020, Art no. 1247.

[21] A. Tandon, B. E. Howard, A. J. Green, R. Elmore, R. Shah, A. Merrick, K. Shockley, K. Ryan, and J.-H. Hsieh. Artificial intelligence (AI)-driven morphological assessment of zebrafish larvae for developmental toxicity chemical screening. *Aquatic Toxicology*, vol. 285, Aug. 2025, Art no. 107415.

[22] D. Čapek, M. Safroshkin, H. Morales-Navarrete, N. Toulany, G. Arutyunov, A. Kurzbach, J. Bihler, J. Hagauer, S. Kick, F. Jones, B.

Jordan, and P. Müller. EmbryoNet: using deep learning to link embryonic phenotypes to signaling pathways. *Nature Methods*, vol. 20, no. 6, pp. 815–823, May 2023.

[23] R. Alshut, J. Legradi, U. Liebel, L. Yang, J. van Wezel, U. Strähle, R. Mikut, and M. Reischl. "Methods for Automated High-Throughput Toxicity Testing Using Zebrafish Embryos," in *KI 2010: Advances in Artificial Intelligence*, 2010, pp. 219–226.

[24] P. Karatzas, G. Melagraki, L.-J. A. Ellis, I. Lynch, D.-D. Varsou, A. Afantis, A. Tsoumanis, P. Doganis, and H. Sarimveis. Development of Deep Learning Models for Predicting the Effects of Exposure to Engineered Nanomaterials on *Daphnia magna*. *Small*, vol. 16, no. 36, Jun. 2020, Art no. 202001080.

[25] S. Inagaki, Y. Kondo, P. Religia, N. Adhitama, Y. Kato, E. Watanabe, and H. Watanabe. Application of deep learning for evaluation of the growth rate of *Daphnia magna*. *Journal of Bioscience and Bioengineering*, vol. 139, no. 5, pp. 384–391, May 2025.

[26] C. Xie. "Development of an Artificial Intelligence-Based Video Monitoring System to Investigate Drug Effects on Daphnia Magna," in *2024 IEEE 3rd International Conference on Computing and Machine Intelligence (ICMI)*, Apr. 2024, pp. 1–4.

[27] S. Cheng, S. Yuan, X. Wu, T. Lei, J. Ji, Y. Yin, Y. Liu, C. Liu, Y. Zhang, and Y. Zhu. Identification of Chemicals Based on Locomotor Tracks of *Daphnia magna* Using Deep Learning. *Environmental Science & Technology Letters*, vol. 10, no. 11, pp. 998–1003, Mar. 2023.

[28] D. Jimenez-Carretero, V. Abrishami, L. Fernández-de-Manuel, I. Palacios, A. Quílez-Álvarez, A. Díez-Sánchez, M. A. del Pozo, and M. C. Montoya. Tox_(R)CNN: Deep learning-based nuclei profiling tool for drug toxicity screening. *PLoS Computational Biology*, vol. 14, no. 11, Nov. 2018, Art no. e1006238.

[29] Y. Zhang, Y. S. Wong, J. Deng, C. Anton, S. Gabos, W. Zhang, D. Y. Huang, and C. Jin. Machine learning algorithms for mode-of-action

classification in toxicity assessment. *BioData Mining*, vol. 9, no. 19, May 2016.

[30] T. Scherr, K. Löffler, M. Böhland, and R. Mikut. Cell segmentation and tracking using CNN-based distance predictions and a graph-based matching strategy. *PLoS One*, vol. 15, no. 12, Dec. 2020, Art no. e0243219.

[31] K. Löffler, and R. Mikut. EmbedTrack–Simultaneous Cell Segmentation and Tracking Through Learning Offsets and Clustering Bandwidths. *IEEE Access*, vol. 10, pp. 77147–77157, Jul. 2022.

[32] F. Hu, S. F. Santagostino, D. M. Danilenko, M. Tseng, J. Brumm, P. Zehnder, and K. C. Wu. Assessment of Skin Toxicity in an *in Vitro* Reconstituted Human Epidermis Model Using Deep Learning. *The American Journal of Pathology*, vol. 192, no. 4, pp. 687–700, Apr. 2022.

[33] N. Toulany, H. Morales-Navarrete, D. Čapek, J. Grathwohl, M. Ünalan, and P. Müller. Uncovering developmental time and tempo using deep learning. *Nature Methods*, vol. 20, no. 12, pp. 2000–2010, Nov. 2023.

[34] L. Gendelev, J. Taylor, D. Myers-Turnbull, S. Chen, M. N. McCarroll, M. R. Arkin, D. Kokel, and M. J. Keiser. Deep phenotypic profiling of neuroactive drugs in larval zebrafish. *Nature Communications*, vol. 15, Nov. 2024, Art no. 9955.

[35] T. Uelwer, J. Robine, S. S. Wagner, M. Höftmann, E. Upschulte, S. Konietzny, M. Behrendt, and S. Harmeling. A survey on self-supervised methods for visual representation learning. *Machine Learning*, vol. 113, Mar. 2025, Art no. 111.

[36] J. Gui, T. Chen, J. Zhang, Q. Cao, Z. Sun, and H. Luo. A Survey on Self-Supervised Learning: Algorithms, Applications, and Future Trends. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 12, pp. 9052–9071, Jun. 2024.

[37] M. Noroozi, and P. Favaro. "Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles," in *Computer Vision - ECCV 2016*, Sep. 2016, pp. 69–84.

[38] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. "A Simple Framework for Contrastive Learning of Visual Representations," in *Proceedings of the 37th International Conference on Machine Learning (PMLR)*, 2020, pp. 1597–1607.

[39] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick. "Masked Autoencoders Are Scalable Vision Learners," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Patter Recognition (CVPR)*, 2022, pp. 16000–16009.

[40] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. "Momentum Contrast for Unsupervised Visual Representation Learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 9729–9738.

[41] X. Chen, H. Fan, R. Girshick, and K. He. Improved Baselines with Momentum Contrastive Learning. *arXiv preprint*, Mar. 2020, Art No. arXiv:2003.04297.

[42] X. Chen, S. Xie, and K. He. "An Empirical Study of Training Self-Supervised Vision Transformers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 9640–9649.

[43] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. A. Pires, Z. D. Guo, M. G. Azar, B. Piot, K. Kavukcuoglu, R. Munos, and M. Valko. "Bootstrap Your Own Latent A New Approach to Self-Supervised Learning," in *34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, 2020, pp. 21271–21284.

[44] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin. "Unsupervised Learning of Visual Features by Contrasting Cluster Assignments," in *34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, 2020, pp. 9912–9924.

[45] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin. "Emerging Properties in Self-Supervised Vision Transformers,"

in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 9650–9660.

[46] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P.-Y. Huang, S.-W. Li, I. Misra, M. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski. DINOv2: Learning Robust Visual Features without Supervision. *arXiv preprint*, Apr. 2023, Art no. arXiv:2304.07193.

[47] O. Siméoni, H. V. Vo, M. Seitzer, F. Baldassarre, M. Oquab, C. Jose, V. Khalidov, M. Szafraniec, S. Yi, M. Ramamonjisoa, F. Massa, D. Haziza, L. Wehrstedt, J. Wang, T. Darcet, T. Moutakanni, L. Sentana, C. Roberts, A. Vedaldi, J. Tolan, J. Brandt, C. Couprie, J. Mairal, H. Jégou, P. Labatut, and P. Bojanowski. DINOv3. *arXiv preprint*, Aug. 2025, Art no. arXiv:2508.10104.

[48] X. Chen, and K. He. "Exploring Simple Siamese Representation Learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 15750–15758.

[49] A. Yamachui Sitcheu, N. Friederich, S. Baeuerle, O. Neumann, M. Reischl, and R. Mikut. "MLOps for Scarce Image Data: A Use Case in Microscopic Image Analysis," in *Proceedings 33. Workshop Computational Intelligence*, Nov. 2023, pp. 169–189.

[50] K. He, X. Zhang, S. Ren, and J. Sun. "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770-778.

[51] A. Krizhevsky. One weird trick for parallelizing convolutional neural networks. *arXiv preprint*, Apr. 2014, Art no. arXiv:1404.5997.

[52] L. McInnes, J. Healy, and J. Melville. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv preprint*, Feb. 2018, Art no. arXiv:1802.03426.

[53] MMPretrain Contributors. *OpenMMLab's Pre-training Toolbox and Benchmark*. (2023). https://github.com/open-mmlab/mmpretrain.

[54] T. Wang, and P. Isola. "Understanding Contrastive Representation Learning through Alignment and Uniformity on the Hypersphere," in *Proceedings of the 37th International Conference on Machine learning (PMLR)*, 2020, pp. 9929–9939.

[55] "Toxicology-testing platform integrating immunocompetent in vitro/ex vivo modules with real-time sensing and machine learning based in silico models for life cycle assessment and SSbD (TOXBOX)," EU Funding & Tenders Portal, [Online]. Available: https://ec.europa.eu/info/funding-tenders/opportunities/portal/screen/how-to-participate/org-details/890666223/project/101138387/program/43108390/details. [Accessed: Sep. 2, 2025].

[56] "TOXBOX: All-in-one toxicology platform for a comprehensive testing of chemicals," TOXBOX Project, [Online]. Available: https://www.youtube.com/watch?v=GXigQY_nZYU. [Accessed: Sep. 7, 2025].

[57] R. Dwivedi, D. Dave, H. Naik, S. Singhal, R. Omer, P. Patel, B. Qian, Z. Wen, T. Shah, G. Morgan, and R. Ranjan. Explainable AI (XAI): Core Ideas, Techniques, and Solutions. *ACM Computing Surveys*, vol. 55, no. 9, pp. 1–33, Jan. 2023.

[58] J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang. "Learning under Concept Drift: A Review," in *IEEE Transactions on Knowledge and Data Engineering*, Oct. 2018, pp. 2346–2363.