Simulating Training Data for Deep Learning Models

Andreas Kamilaris

1. Introduction

Deep learning (DL) constitutes a recent, modern technique for image processing and data analysis with large potential [1-2]. DL belongs to the machine learning (ML) computational field and is similar to artificial neural networks (ANN). DL extends ML by adding more "depth" (complexity) into the model, transforming the data using various functions that allow data representation in a hierarchical way, through several abstraction levels. DL allows larger learning capabilities and thus higher performance and precision. DL consists of various different components (e.g. convolutions, pooling layers, fully connected layers, gates, memory cells, activation functions, encode/decode schemes etc.), depending on the network architecture used. DL models are adaptable to a wide variety of highly complex challenges [3], not necessarily related to computer vision. DL is divided in discriminative and generative models [4]. The former is about predictions/classifications, and the latter about synthesis/generation of data similar to the input datasets.

An advantage of DL is the reduced need of feature engineering (FE). Previously, traditional approaches for image classification were based on hand-engineered features, whose performance affected heavily the results. FE is a complex, time-consuming process, depending on experts' knowledge, not generalizing well [5]. DL does not require FE, locating important features automatically through training (feature learning, FL). FL is the automatic feature extraction from raw data, with features from higher levels formed by composing lower level features [6].

Although DL does not require FE, it still needs appropriate datasets as input in DL models during learning. These datasets need to be large, to allow DL models to learn the problem elaborately, and expressive, to capture the variation of classes/features that need to be classified/predicted at the model output. An existing problem is the limited availability of such appropriate datasets. This limitation makes DL models sometimes difficult to generalize and to learn the problem well, towards high precision.

A recent possibility is the development of simulated datasets to train DL models, which are properly designed to solve real-world problems. The use of simulated data to train DL models is promising, with early attempts in agriculture indicating positive outcomes (see Section 2). Models are trained using synthetic images, and then tested on real ones. The SimDeepL project plans to study the general potential of the approach of using simulated data to train DL models. It aims to understand how it works, consider its advantages and disadvantages, constraints and limitations, and range of problems over which it can be applied.

1 Department of Computer Science, University of Twente, The Netherlands. Email: a.kamilaris@utwente.nl
2. Related Work

A problem in discriminative DL is the limited availability of datasets which are large and expressive enough. The more complicated the problem, the more data is required [7]. For example, problems involving a large number of classes [8-10] and/or small variation among the classes [10-12], require a large number of input images to train their models. Furthermore, variation between classes is necessary for the DL models to differentiate features and characteristics, and to perform accurate classifications. Hence, accuracy is positively correlated with class variation. As the survey of Kamilaris et al. indicates [13], when different datasets were used for testing than the ones used for training, the model had difficulties to generalize well. A technique called data augmentation [14], is used to enlarge artificially image datasets. This technique is important for problems where only small datasets are available for training DL models. Transformations are label-preserving, and include rotations, dataset partitioning/cropping, scaling, transposing, mirroring, translation and perspective transform. Augmentation could increase the dataset by 4-5 times, but since the transformed images are based on the initial dataset, there are upper boundaries to the learning capabilities of the model. The problem is bigger in datasets targeting anomaly detection [15], since anomaly-related data is usually only a very small percentage of the total dataset.

In cases of limited datasets, the technique of using simulated datasets to train DL models could be an alternative, promising solution. Early initiatives targeted the agricultural domain, as follows:

1. Detecting and classifying weed species and maize in fields [16]. The authors overcame the plant foliage overlapping problem by simulating top-down images of overlapping plants on soil background. They also varied the HSV channels adding random shadows in simulated images. The trained network was then capable to distinguish weeds from maize even in overlapping conditions.
2. Predicting number of tomatoes in the images [17]. The authors used color circles to simulate the background and tomato plant/crops.
3. Identifying roots from soil [18]. The authors added simulated roots to soil images coming from X-ray tomography, helping the model to learn the problem well, which was that soil/root contrast was low at the initial real dataset.

In these cases, synthetic/simulated datasets allowed the models to generalize and adapt to real-world problems more easily. The papers above claimed to achieve satisfactory precision, using different performance metrics. Simulated data allowed to address classical problems that hinder high-precision classification. For example, in the problem of fruit counting, datasets tend to suffer from high occlusion, depth variation, and uncontrolled illumination, including high color similarity between fruit/foliage [19-20]. By means of simulated data in [17], the model learned explicitly to count, being robust to occlusion, variation, illumination and scale. Moreover, identification of weeds faces issues with respect to lighting, resolution, and soil type, and small variation between weeds and crops [10], [21]. The simulated data helped the model to address these issues well in [16].

Finally, it is worth mentioning the importance of simulated data in deep robotic learning [22], where a combination of real and simulated data provides improved performance.
3. Methodology

The general methodology of the proposal is illustrated in Figure 1. DL models are trained with simulated data, and then tested with real-world data. The precision/accuracy results are compared with the state-of-the-art related work, and this comparison is given as feedback to the creation process of the simulated datasets, to become more detailed and complete. The main focus of DL-related challenges is on classification or counting problems (e.g. fruit counting). Thus, the research performed targets discriminative DL models, although concepts from generative DL could be used for the creation of the simulated datasets.

Our approach in simulated dataset design would be to understand how DL models perform classification, based on the existing real-world datasets. To achieve this, we take advantage of the work in [23], which allows to visualize what happens inside DL models, i.e. which aspects/characteristics of the image are the ones that trigger the final classification. These characteristics could then be used to better design the simulated datasets, emphasizing on these aspects when creating the simulated images.

Our simulated datasets are created by means of Python, together with the Python Imaging Library\(^2\) and OpenCV\(^3\). These libraries allow to combine graphics creation, together with programming code and computer logic, using code in order to create dots, lines, rectangles, polygons, circles, ellipses and combinations, allowing to add color, transparency, borders and outlines, but also to include filters such as “Gaussian Blur”, smoothen the image, enhance the edges etc. Example images from the melanoma simulated dataset are presented in Section 4.

---

\(^2\) Python Imaging Library. [https://pypi.python.org/pypi/PIL](https://pypi.python.org/pypi/PIL)

\(^3\) OpenSource Computer Vision (OpenCV). [https://pypi.python.org/pypi/opencv-python](https://pypi.python.org/pypi/opencv-python)
We then compare the precision results by using the simulated datasets, vs. the precision results by using real-world datasets for training, using in both cases a sub-set of the real-world dataset for testing. Our goal is to reach (and if possible increase) the overall accuracy/precision of the DL model by means of the simulated dataset used in training. Table 1 below explains this process. The split of the real-world dataset in 85% training and 15% testing has been selected as it has provided the best results in related work [13].

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Training Data</th>
<th>Testing Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulated case</td>
<td>Simulated datasets</td>
<td>15% of real-world dataset</td>
</tr>
<tr>
<td>Real-world case</td>
<td>85% of real-world datasets</td>
<td>15% of real-world datasets</td>
</tr>
</tbody>
</table>

**Tab. 1.** Comparison between simulated and real-world-based DL models.

The performance metrics used generally during the evaluation of our approach are the following: Classification Accuracy (CA); Precision; F1 score; and Root Mean Square Error (RMSE). These metrics are used because they are commonly used in DL generally. For the case study described in Section 4, only CA has been used. It is important to use the same metric when comparing the two approaches.

### 4. Case Study: Melanoma

Skin cancer is a major public health problem, with over 5 million newly diagnosed cases in the United States each year. Melanoma is the deadliest form of skin cancer, responsible for over 9,000 deaths each year. Images of melanoma cancer have some particular features [24], which could be simulated and be used to train a DL model using simulated datasets. There are also available datasets for melanoma images available, which could be used to compare the simulated-based ones, with the real-based ones.

#### 4.1 Simulated Dataset Creation

From the available datasets, we selected the one provided by the International Skin Imaging Collaboration (ISIC) 2017 challenge [25]. This dataset contains two different classes:

1. Melanoma — malignant skin tumor, derived from melanocytes (melanocytic)
2. Benign skin tumor — Either derived from melanocytes (melanocytic), called *Nevus*, or derived from keratinocytes (non-melanocytic), called *Seborrheic keratosis*.

In regards to the simulated datasets, we considered the following ABCDE general rules for melanoma:

- **Asymmetry:** If you draw a line through this mole, the two halves will not match, meaning it is asymmetrical, a warning sign for melanoma.
- **Border:** Borders of an early melanoma tend to be uneven. The edges may be scalloped or notched.
- **Color:** Having a variety of colors is another warning signal. A number of different shades of brown, tan or black could appear. A melanoma may also become red, white or blue.
- **Diameter:** Melanomas usually are larger in diameter than the eraser on your pencil tip (¼ inch or 6mm), but they may sometimes be smaller when first detected.

- **Evolving:** Be on the alert when a mole starts to evolve or change in any way. When a mole is evolving, see a doctor. Any change — in size, shape, color, elevation, or another trait, or any new symptom such as bleeding, itching or crusting — points to danger.

The ABCD rules were relatively easy to include in the simulated dataset design, but for the E(volving) rule, one would require a different architecture (i.e. RNN, LSTM) and overall design. As the ISIC 2017 dataset focused on images of melanoma and seborrheic keratosis/Novus, we studied more specifically the particular characteristics and differences between these skin lesion types, before starting the design of the simulated dataset. These characteristics are listed in Table 2.

<table>
<thead>
<tr>
<th>Skin lesion</th>
<th>Different characteristics</th>
<th>Similarities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Melanoma</td>
<td>Sides that do not match in size/shape, fuzzy border or ragged/blurred edges, variety of colors within the same mole</td>
<td>Growthhs can be brown or black, growths can vary in size, growths can appear anywhere on the body.</td>
</tr>
<tr>
<td>Seborrheic keratosis/Novus</td>
<td>Round/oval shaped, light tan in color, waxy surface, often appear in groups of two/more</td>
<td></td>
</tr>
</tbody>
</table>

Tab. 2. Comparison between melanoma and seborrheic keratosis/Novus.

Based on the above, we designed our simulated datasets for the two classes of the problem. Example images of the dataset are depicted in Figure 2.

**Fig. 2.** Simulated images of melanoma (top row) and seborrheic keratosis/Novus (bottom row).

### 4.2 Implementation and Analysis

We simulated 100,000 images, while the original ISIC 2017 challenge dataset contained 2,000 images (85% of these used for training, i.e. 1,700 images). The distribution of images in classes during training is listed

---

in Table 3. Regarding testing, a 15% subset of the ISIC 2017 challenge dataset has been used, i.e. 300 images (60 images of melanoma and 240 images of seborrheic keratosis/Novus).

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Melanoma</th>
<th>Seborrheic keratosis/Novus</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISIC 2017 Challenge dataset</td>
<td>314</td>
<td>1386</td>
<td>1,700</td>
</tr>
<tr>
<td>Simulated dataset</td>
<td>50,000</td>
<td>50,000</td>
<td>100,000</td>
</tr>
</tbody>
</table>

**Tab. 3.** Distribution of datasets/images in classes during training.

We selected ResNet50 as our convolutional neural network (CNN) model [26], as it is one of the fastest ones, with accuracy highly comparable with Inception-v4, which is slightly slower, according to [27]. We used the class provided by Keras/TensorFlow, pre-trained with the ImageNet dataset. In the ISIC 2017 dataset, we used also data augmentation [14]. The model was trained for 10 epochs in the ISIC dataset case and 4 epochs in the simulated dataset case, and the results are presented in Table 4.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Classification Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISIC 2017 Challenge dataset</td>
<td>0.769</td>
</tr>
<tr>
<td>Simulated dataset</td>
<td>0.775</td>
</tr>
</tbody>
</table>

**Tab. 4.** Classification accuracy of the deep learning models, trained with different datasets.

The CA achieved by ResNet50 for the ISIC dataset is a bit higher than the one recorded in [28]. The CA in both scenarios tested is very similar, with the simulated dataset-based case slightly outperforming the real dataset-based one. This indicates that it is indeed possible to use simulated data to train DL models.

5. **Discussion**

This work constitutes an innovative effort in the research area of DL, as it is one of the first initiatives -to our knowledge- that examines how simulated data can be used for training DL models, beyond some basic, intuition-based early efforts, which indicated positive results. Besides the aforementioned research efforts employing simulated data [16-18, 22], there do not exist other efforts using this practice. It is a new possibility for the DL community, with much unexplored ground for research. Design and creation of the simulated datasets might not be a trivial task, depending on the domain, as it sometimes requires complete understanding of the features that characterize some class. This was the case for melanoma vs. seborrheic keratosis/Novus, where we had to study carefully the characteristics and differences between these skin lesions. A collaborative effort between real and simulated datasets could also be made, i.e. to use the real-world datasets to extract the features that distinguish the classes, as understood by the DL model, and then exploit these features in order to design simulated datasets that would be used in combination with the real ones, to enhance the overall predictions and accuracy.
We expect that the proposed technique, if successful, will change radically the research area of discriminative DL, especially in classification/counting problems. Applications could be found in various research domains and scientific disciplines, such as agriculture, life sciences, microbiology, earth sciences etc. This approach would be extremely useful for robotics [22], where computer vision is involved. It could improve operation and accuracy of automatic robots collecting crops, removing weeds or estimating yields of crops. It could also be used in disaster monitoring and surveillance where remote sensing (i.e. satellites or unmanned aerial vehicles) is used to identify events of interest (e.g. disasters, violence incidents, land cover mapping, effects on climate change etc.). A more futuristic application could be in microbiology for human/animal cell counting [19]. Although the main influence of this work is in discriminative DL, it could also change the direction of research in generative DL, leading to more scientists experimenting with generative DL towards creating more realistic, simulated datasets to train DL models for real-world problems. This paper describes on-going work, and we are in the process of designing scenarios at which simulated data is used to train DL models for the identification of disasters such as flooding, earthquakes, fire etc. This should be feasible, because all these disasters have clear, specific characteristics [29]. Flooding has large percentage of water in the image, earthquakes cause damages in buildings, while fire has some characteristic color but also it can be inferred by smoke.

6. Conclusion

This paper describes preliminary work in the innovative effort of simulating training data for facilitating the learning procedure of deep learning models. The general concept and methodology are described, and preliminary results based on a case study on melanoma have been presented. The proposed technique constitutes a new possibility for the DL community, with unexplored ground for research, and it has the potential to change radically the research area of discriminative DL, especially in classification problems.

References


