Rudi: Lightweight image converter and dataset augmentor

Petro Liashchynskyi

Department of Computer Engineering,
Ternopil National Economic University,
Ternopil 46003, Ukraine

1 Purpose of the software

I present Rudi, a Python tool that simplifies the dataset preparation for training a convolutional neural network or other things that need images as input data. The tool contains a set of practical generalized tools, which can wrap common operations like image resizing, flipping, zooming, distortion and others. Presented tool can be helpful with extending the dateset when the user does not have enough of data. After installation Rudi, the tool can be called by typing *rudi* in the terminal window.

2 Background

Creating a train and test dataset of images of adequate size is a very non-trivial task. And this is not about the technical difficulties of collecting and storing a million images, but about the perennial situation when you have one and a half pictures in the first stage of system development. In addition, it should be understood that the composition of the training set can affect the quality of the resulting recognition system more than all other factors.

Deep artificial neural networks require a large corpus of training data in order to effectively learn, where collection of such training data is often expensive and laborious. Data augmentation overcomes this issue by artificially inflating the training set with label preserving transformations. Recently there has been extensive use of generic data augmentation to improve Convolutional Neural Network (CNN) task performance [3][5].

Data augmentation is a technique to artificially create new training data from existing training data. This is done by applying domain-specific techniques to examples from the training data that create new and different training examples. Image data augmentation is perhaps the most well-known type of data augmentation and involves creating transformed versions of images in the training dataset that belong to the same class as the original image.

To create a larger base of training samples, data augmentation techniques are found effective to enlarge the training datasets and lead to better performance. Traditional data augmentation techniques include traditional transformation measures such as image flipping and rotations, random cropping, random scaling, center zooming, brightness and sharpness adjusting and noise mixing [4].

3 Summary

Modern deep learning algorithms, such as the convolutional neural network (CNN), is able to learn features that are invariant to their location in the image. Nevertheless, augmentation can further aid in this transform invariant approach to learning and can aid the model in learning features that are also invariant to transforms such as flipping, rotating, light levels in photographs, contrast and much more. Image data augmentation is typically only applied to the training dataset, and not to the validation or test dataset [1].

Also, for training a neural network that accepts images as input data it needs to convert images in dataset to the same aspect ratio, extension, etc. Rudi will take care of it for a user. Rudi is a command line tool for dataset augmentation and transformation.

The software Rudi is built upon precursor algorithms and software for image operations and data augmentation.

Rudi is implemented in Python. It takes as input a root directory of dataset images and a few additional options and outputs converted images to output directory in a root.

Rudi tool provides two main commands:

- rudi convert
- rudi augment

The first is to convert the dataset of images, with its typical interface, which converts data to specific format (**file extensions jpg and png**) and size defined by a user. The second is to augment the entire dataset with operations, such as **flipping, rotating, zooming, random distortion and skewing.**

4 Examples

This tool was used for expanding the dataset of biomedical images [2] from under 100 to over 1500 images (see Figure 1).

As described by [2] this approach allowed author to achieve *95%* accuracy on *histological images dataset* (5 classes) (see Figure 2).

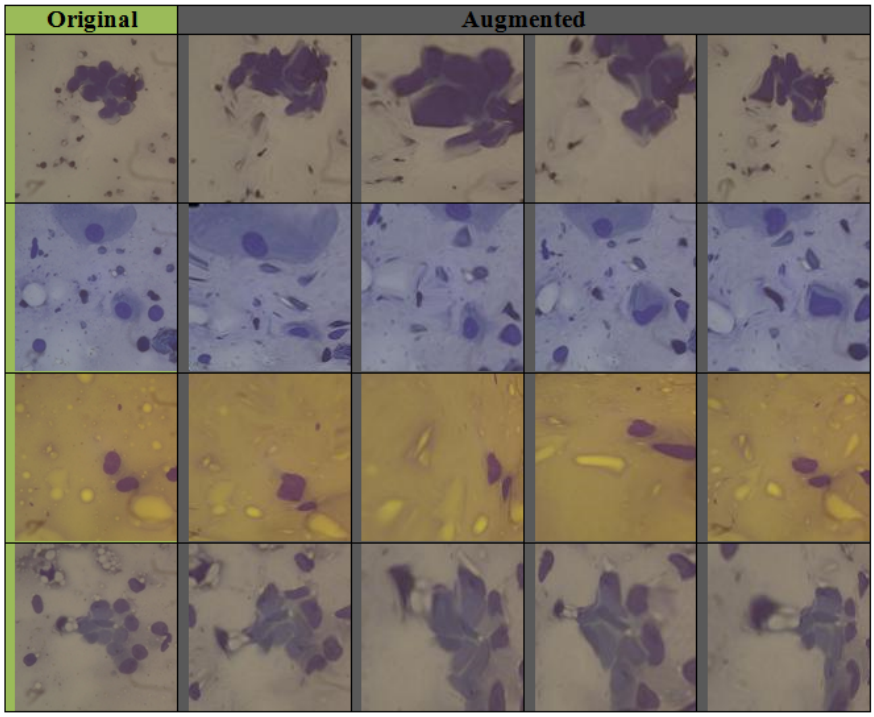

Figure 1: Example images of the original and augmented dataset.

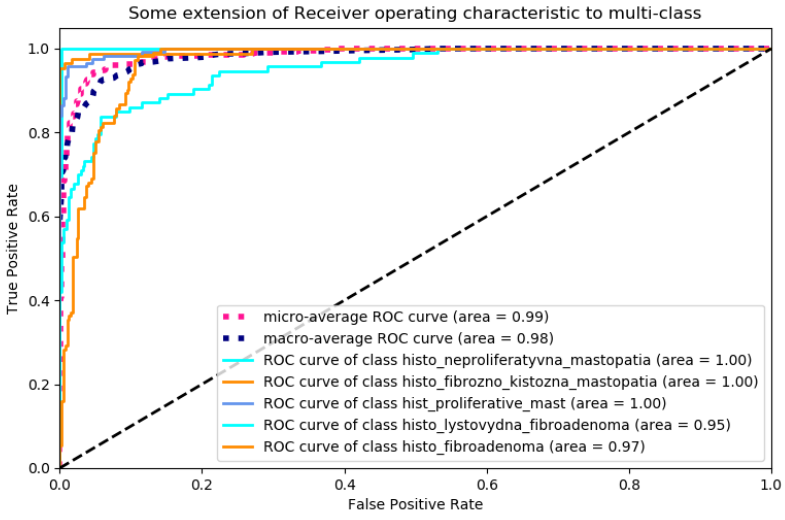

Figure 2: ROC crurves for CNN classifier for histological images.

References

[1] Brownlee Jason. *How to Configure Image Data Augmentation in Keras*. 2019. URL: https://machinelearningmastery.com/how-to-configure-image-data-augmentation-when-training-deep-learning-neural-networks/.

[2] Petro Liashchynskyi. "CNN-based biomedical image generation". In: *Intellectual Computer Systems and Networks*. 2019, p. 55. URL: https://drive.google.com/file/d/1FPJPB5Zn9WKTB_PrT-8gI6M5hVgTmmeb/view.

[3] Taylor Luke and Geoff Nitschke. "Improving Deep Learning using Generic Data Augmentation". In: *arXiv preprint arXiv:1708.06020* (2017). URL: https://arxiv.org/abs/1708.06020.

[4] Agnieszka Mikolajczyk and Michal Grochowski. "Data augmentation for improving deep learning in image classification problem". In: May 2018, pp. 117–122. DOI: 10.1109/IIPHDW.2018.8388338.

[5] Connor Shorten and Taghi Khoshgoftaar. "A survey on Image Data Augmentation for Deep Learning". In: *Journal of Big Data* 6 (Dec. 2019). DOI: 10.1186/s40537-019-0197-0.