Random Pixel Selection through Image Cropping for Data Augmentation and Classification

1st Boudouh Nouara
LESIA Laboratory, University of Mohamed Khider
Biskra, Algeria
nouara.boudouh@univ-biskra.dz

2nd Mokhtari Bilal
Dept. of computer science, University of Mohamed Khider
Biskra, Algeria
bilal.mokhtari@univ-biskra.dz

***Abstract*—The lack of data constitutes an obstacle to getting high accuracy during training Deep Neural Networks. One of the strategies addressing the challenge of this problem is data augmentation. Through this paper, we propose a new method based on the random selection of pixels by cropping some rows and columns from a given image to create new images with a reduced number of pixels. The proposed method allows the creation of new images that differ from the original image, by preserving a set of pixels selected from designated lines and columns. This assures enriching and diversifying the original dataset while trying to capture the regions of interest and dismissing the other regions, which leads to better generalization of the training model. The accuracy of the proposed method on ResNet50 Network applied to the Kaggle Cat VS Dog dataset achieves 88.93% with 125002 samples, whereas the accuracy when applied to the original dataset with 25002 was 85.68%.**

***Index Terms*—Image classification, data augmentation, pixel selection, and image cropping.**

I. Introduction

Deep Learning (DL) and machine learning (ML) approaches are now widely used, as evidenced by the fact that they are applied in a wide variety of fields. [1]. Despite enormous progress, however, a data shortage has always been a problem because it is difficult to obtain data, especially in practical areas like face emotion identification. [2]. Many strategies have been proposed to address this problem. Starting with transfer learning, it makes use of the knowledge gained while solving one problem and applying it to a different but related problem [3]. Dropout is another strategy in which a new approach is relatively used to train neural networks, it depends on randomly dropping out neurons during CNN training [4]. However, one of the most used strategies is data augmentation, it is not related to CNN, but the data. It has emerged as a data enrichment and enhancement method and can be presented as the process of augmenting an existing data collection using several methods, also it can be used to increase the dataset's size [5]. It can amplify other kinds of data, such as sound [6], or Natural Language Processing NLP [7]. As mentioned in [8], [9], several methods have been proposed and are generally divided into two categories: traditional and DL-based approaches. The first concerns image manipulations and geometric transformations including rotation, translation, scaling, etc. The second is based on Generative Adversarial Networks GANs.

II. Proposed method

Our proposed method is based on the selection of a set of pixels belonging to some rows and columns of a given image for creating new images. It allows to cropping of some lines and columns from a given image using filters generated according to a random percentage (percentage of the preserved pixels from the original image to create a new one). The goal is to enrich the dataset by adding new images that differ from the original images, which allows diversifying the data, consequently providing overfitting. As mentioned above, in the random erasing data augmentation method proposed by Zhong et al. [10], some pixels are erased by changing their values, but the image size does not change. Moreover, in the cutout method proposed by DeVries et al., [11], an entire region may be masked, which may allow important information to be lost. However, our approach just crops some rows and columns of pixels, not necessarily adjacent. This allows the preservation of significant information from the different regions of the image as well as the creation of smaller images (reducing the image size), which, of course, has its advantages (decreased memory space occupied by the augmented dataset). The selection of some pixels allows us to find the most important regions in the images (representative regions), which allows for improving the classification process. Note that the proposed method depends on two factors, which have a significant impact on the obtained results, they are the number of selected pixels and the position of selected rows and columns. Thus, the proposed method generates images with different sizes obtained by selecting randomly the positions and the numbers of rows and columns to be preserved. The overall architecture of the proposed method is presented in Figure 1, it goes through the following stages:

1) For each image in the original dataset, we generate a set of different filters.
2) Apply the generated filters for a given image to obtain as much as filters as new images.
3) Both original and augmented images are used to train CNN.

Therefore, for each given image from the original dataset, we generate a number of filters (compatible with the number of new images to be created from the original image). The filter is a matrix that has the same dimension as the given image,

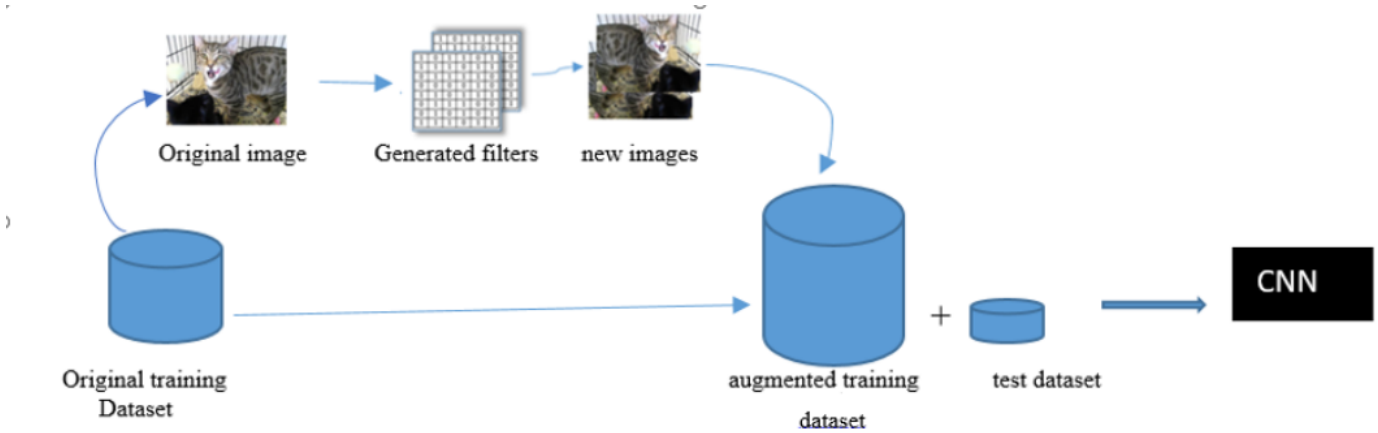

Fig. 1. Architecture of the proposed method used to generate augmented images.

generated randomly and initialized to '0'. We select in the filter a certain number of rows and columns calculated according to equation 1 and equation 2 and then change the values of a random set of pixels to 1 (Algorithm 1). Those rows and columns present the preserved pixels which will form later the new image. The number and position of the selected pixels are different in each filter. Therefore, the filter contains some rows and columns with values of 0 (those rows and columns of pixels will be cropped in the original image, see Figure 2). Values 1 in the filter correspond to pixels that will be preserved. To create a new image, we must transform the given image into a matrix of pixels, then we browse simultaneously the image matrix and the creating filter. Each pixel whose position corresponds to 1 in the filter will be selected from the given image to be preserved in the new image matrix. Finally, the resulting image matrix will be converted into an image to be saved.

Each filter produces a new image (we create many filters for each original image according to a given probability of selecting rows and columns). The resulting images will be used later with the original training images to train the convolution nerol network (CNN). Note that in the test phase, only the original images are used to validate the results.

In fact, the number of taken pixels from the original image represents the new resolution of the created image (The number of rows and columns of the new image matrix is calculated according to equations 1 and 2). Let be:

- n the number of lines, and m the number of columns in the original image.
- L_i and C_j are respectively the indices of lines and columns in the original image.
- F is a matrix $n \times m$ called filter, such that all its values are initialized to 0.
- μ_1 and μ_2 present respectively the number of lines and columns of the target image (equations 1 and 2)
- P_1 and P_2 represent respectively the probability for a line or a column to be selected.
- N presents the new resolution of the target image.

Therefore:

$$\mu_1 = \lceil P_1 n \rceil \tag{1}$$

$$\mu_2 = \lceil P_2 m \rceil \tag{2}$$

$$N = \mu_1 \times \mu_2 \tag{3}$$

Let be X and Y discrete random variables, so the probability functions $f(X)$ and $f(Y)$ can be written as follows:

$$f(L_i) = P_1(X = L_i) \tag{4}$$

and:

$$f(C_j) = P(Y = C_j) \tag{5}$$

Such as:

$$\sum_{i=1}^{n} f(L_i) = 1, \sum_{i=1}^{m} f(C_i) = 1$$

Therefore:
The value of all pixels belonging to the selected lines and columns in F is set to 0. It represents the pixel to be cropped from the original image.

The use of the filter F allows every time to obtain a different new image from the same original one. Algorithm 1 shows the steps of creating a filter for a given image, and figure 2 shows how to apply the creating filter, by selecting pixels corresponding to number 1 in a given image, to create a new image (Pixel selected from the original image corresponds to the pixel with value 1 in the filter).

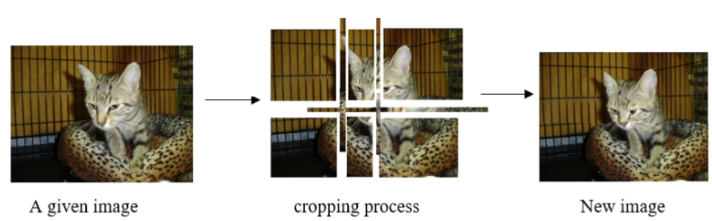

Fig. 2. Example of obtained image applying a given filter.

TABLE I
RESULT OF TRAINING DATASET ACCORDING TO A DIFFERENT NUMBER OF EXAMPLES.

Test	Train dataset size	Description	Accuracy	Error
01	x	The original dataset	85.68%	0.6834
02	4x	The augmented dataset	88.93%	0.3479

Algorithm 1. Creating a new filter F
Designate lines L_i and columns C_j using equations 4 and 5
Calculate μ_1 the number of lines in the target image
Calculate μ_2 the number of its columns
Generate a random vector VRn containing indices of selected rows
Generate a random vector VCn containing indices of selected columns
while $i < \mu_1$ **do**
  $VRn[i] = L_i$ /* i presents the line number of the selected pixel */
  while $j < \mu_2$ **do**
    $VCn[j] = C_j$ /* j presents the column number of the selected pixel */
    F[i][j] = 0 /* Select the pixel to remove */
  end while
end while

Considering different indices and numbers of rows and columns selected using the filter, which is generated randomly in each iteration (each filter is different), allows for diversification of the resulting images. In figure **??**, we can see that the newly obtained images are different because of the use of different filters.

Algorithm 2 shows the steps of getting new images by applying different filters. Figure 2 shows an example of cropping two lines and one column using Algorithm 2 to obtain a new image.

Algorithm 2. Creating new images by applying a given filter F.
Open the image as an array
Getting image dimensions n and m
Creating a new filter F using Algorithm 1
k = 0
for $i = 1$ to n **do**
  h = 0 /* k and h are the respectively the indices of lines and columns in the new image*/
  for $j = 1$ to m **do**
    if F[i][j] == 1 **then**
      newImage[k][h] = image[i][j] /* preserve the same pixels from the given image */
      h=h+1
    end if
  end for
  K=k+1
end for
Save newImage /* add it to the original dataset */

III. RESULTS AND DISCUSSION

Because it is challenging to distinguish between cats and dogs, we used the Database Cats Vs. Dogs from Kaggle to test the usefulness of the proposed strategy. In computer vision, the cats vs. Dogs dataset is frequently used to classify images as containing either cats or dogs. It contains 25002 images, which we divided into 20000 training images (10,000 images in each class), and 5002 testing images (2501 in each class). It is separated into two classes (cats and dogs). In this research, we evaluate and contrast the training results of the ResNet50 Convolution Neural Network using the original dataset, and the augmented dataset by our proposal.

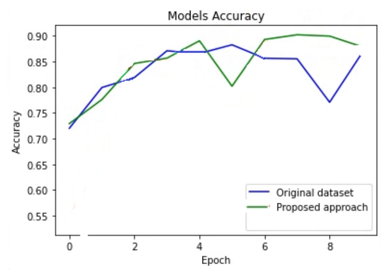

Fig. 3. Accuracy recorded from utilizing the cats vs. dogs dataset (both the original database and the augmented dataset) to train the ResNet50 model.

Table I shows the obtained results using our proposed methods compared to the original results. According to the obtained results, we notice that when the size of data increased by 4X (the original size is X) the accuracy increased by 3.25% (the accuracy with the original database was 85.68 %, while with the size increases by 4X, the accuracy has risen to 88.93%). When the error was reduced by 0.3355. thus results show that the proposed method produces diverse images that are different from each other, by selecting a subset of pixels from the original image and enriching the original dataset. As shown in the above curves Figures 3, the training's accuracy of the augmented dataset increases steadily more than the accuracy of the original dataset. This shows that the model is able to generalize from the augmented data and successfully capture the features in the images. It proves the high quality of the generated image using the proposed approach.

IV. CONCLUSION

In this paper, we proposed a data augmentation method based on cropping rows and columns leading to selecting a subset of pixels from a given image to generate a new set of images, which can be used to enrich the dataset and avoid overfitting. The obtained results show that the results are enhanced when injecting these new images into the original dataset. In future works, we are planning to use optimization methods such as backpropagation to find the optimal images among the obtained images when applying filters, which can improve the classification results.

REFERENCES

[1] Shinde, P. P., and Shah, S. (2018, August). A review of machine learning and deep learning applications. In 2018 Fourth International Conference on computing communication control and Automation (ICCUBEA) (pp. 1-6). IEEE.

[2] Yi, W., Sun, Y., and He, S. (2018, August). Data augmentation using conditional GANs for facial emotion recognition. In 2018 Progress in Electromagnetics Research Symposium (PIERS-Toyama) (pp. 710-714). IEEE.

[3] Zhuang, Fuzhen, et al. "A comprehensive survey on transfer learning." Proceedings of the IEEE 109.1 (2020): 43-76.

[4] Baldi, P., and Sadowski, P. J. (2013). Understanding dropout. Advances in neural information processing systems, 26.

[5] Kaur, P., Khehra, B. S., and Mavi, E. B. S. (2021, August). Data augmentation for object detection: A review. In 2021 IEEE International Midwest Symposium on Circuits and Systems (MWSCAS) (pp. 537-543). IEEE.

[6] Ko, B. Y., Nam, H., Kim, S. H., Min, D., Choi, S. D., and Park, Y. H. (2022). Data Augmentation and Squeeze-and-Excitation Network on Multiple Dimension for Sound Event Localization and Detection in Real Scenes. arXiv preprint arXiv:2206.12059.

[7] Morris, J. X., Lifland, E., Yoo, J. Y., Grigsby, J., Jin, D., and Qi, Y. (2020). Textattack: A framework for adversarial attacks, data augmentation, and adversarial training in nlp. arXiv preprint arXiv:2005.05909

[8] Kaur, P., Khehra, B. S., and Mavi, E. B. S. (2021, August). Data augmentation for object detection: A review. In 2021 IEEE International Midwest Symposium on Circuits and Systems (MWSCAS) (pp. 537-543). IEEE.

[9] Shorten, C., and Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. Journal of big data, 6(1), 1-48.

[10] Zhong, Zhun, et al. "Random erasing data augmentation." Proceedings of the AAAI conference on artificial intelligence. Vol. 34. No. 07. 2020.

[11] DeVries, Terrance, and Graham W. Taylor. "Improved regularization of convolutional neural networks with cutout." arXiv preprint arXiv:1708.04552 (2017).