ITC 4/52

Information Technology and Control

Vol. 52 / No. 4 / 2023

pp. 996-1009

DOI 10.5755/j01.itc.52.4.33828

CAugment: An Approach to Diversifying Datasets by Combining Image Processing Operators

Received 2023/04/13 | Accepted after revision 2023/08/14

HOW TO CITE: Gao, W. (2023). CAugment: An Approach to Diversifying Datasets by Combining Image Processing Operators. *Information Technology and Control*, 52(4), 996-1009. https://doi.org/10.5755/j01.itc.52.4.33828

CAugment: An Approach to Diversifying Datasets by Combining Image Processing Operators

Wuliang Gao

School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, 800 Dongchuan Road, Minhang District, Shanghai, China; phone: +86 13524956692; e-mail: wlgaomail@163.com

Corresponding author: wlgaomail@163.com

In deep learning, model quality is extremely important. Consequently, the quality and the sufficiency of the datasets for training models have attracted considerable attention from both industry and academia. Automatic data augmentation, which provides a means of using image processing operators to generate data from existing datasets, is quite effective in searching for mutants of the images and expanding the training datasets. However, existing automatic data augmentation techniques often fail to fully exploit the potential of the data, failing to balance the search efficiency and the model accuracy. This paper presents CAugment, a novel approach to diversifying image datasets by combining image processing operators. Given a training image dataset, CAugment is composed of: 1) the three-level evolutionary algorithm (TLEA) that employs three levels of atomic operations for augmenting the dataset and an adaptive strategy for decreasing granularity and 2) a design that uses the three-dimensional evaluation method (TDEM) and a dHash algorithm to measure the diversity of the dataset. The search space can be expanded, which further improves model accuracy during training. We use CAugment to augment the CIFAR-10/100 and SVHN datasets and use the augmented datasets to train the WideResNet and Shake-Shake models. Our results show that the amount of data increases linearly along with the training epochs; in addition, the models trained by the CAugment-augmented datasets outperform those trained by the datasets augmented by the other techniques by up to 17.9% in accuracy on the SVHN dataset.

KEYWORDS: Data Augmentation, Image Processing, Deep Learning.

1. Introduction

Deep learning relies on both models and data, both of which have undergone intensive development over the past decade. In deep learning, a model serves as the foundation for learning. For example, convolutional neural networks (CNNs) have become a cornerstone of computer vision (CV), while long short-term memory (LSTM) [12] has revolutionized natural language processing (NLP). The learning process consists of two main steps, training and testing. The former refers to a subprocess in which the engineers train the model using labeled/unlabeled data. The later is to evaluate the performance of the model.

Researchers often focus on designing and tuning models rather than on the quality and sufficiency of the data for training the models. As the model complexity increases, a large amount of effort needs to be spent on preparing the datasets for model training. However, the data may still not be sufficient or suitable for model training, which makes the model training less effective.

While much attention has been given to designing effective models and techniques for model tuning, the quality of the data used for training is often overlooked. As models of parameters become more and more complex, it can be time-consuming and challenging to prepare high-quality datasets for model training. To tackle with the challenge, data augmentation has been developed to expand the dataset by exploring existing data as much as possible, or by generating data with additional features. For example, in CV, data augmentation methods typically fall into two categories, either performing various transformations on the image using operators, or generating new data through generative adversarial network (GAN) [8] models. Techniques such as meta-learning, adversarial training and neural style transfer (NST) [14] may also be used. They are sometimes combined with additional models for data processing, which often improves model accuracy.

Data augmentation through image processing operators is a typical, but effective technique in increasing the amount of image data for model training. However, the sheer number of operators make it difficult to choose the most suitable operator for data augmentation. Despite this, research has shown that the use of multiple operators can significantly augment data and improve the accuracy of deep learning models. Unfortunately, finding an optimal combination of operators can be time-consuming, as each operator contains multiple parameters, some of which may be discrete or continuous. This requires human experiences and a considerable amount of time to tune the parameters, making the process inefficient. To overcome this challenge, data augmentation may employ a heuristic to combine multiple operators in a more efficient manner.

Automatic data augmentation automates the search for suitable operators and their combinations. A search space is thus defined by discretizing the parameters of each operator and composing a number of operators, the parameter intensity and the selection of operators. Different search algorithms can be used to find heuristics leading to a highly accurate model. The efficiency of the search algorithm is another key factor, which makes a balance between the algorithm efficiency and the model accuracy. By employing these heuristics, automatic data augmentation can significantly improve the accuracy of the model, making deep learning more effective for real-world applications.

However, automatic data augmentation can sacrifice data diversity, as it may fail to discover new information in the data. Existing automatic data augmentation techniques often use a limited number of operators, which reduces the size of the search space and simplifies the search problem. It allows for relatively efficient search algorithms but may lead to less diverse data. As the number of operators increases, the data change more substantially but the search space also grows exponentially. To fully leverage the potential of the data, it is important to consider not only the efficiency and accuracy of the algorithm but also its ability to explore the whole search space of image mutants.

This paper presents CAugment, a novel method for diversifying datasets by combining various image processing operations. CAugment leverages a *three-level evolutionary algorithm* (TLEA), in which each level involves a specific heuristic. It allows the algorithm to continuously adapt to the search space during training. CAugment also utilizes the TDEM and a dHash algorithm [40] to compare the data generated throughout the training process.

Our research contributions are listed as follows:

- CAugment employs the TLEA to expand image datasets while maintaining both search efficiency and model accuracy.
- CAugment leverages the TDEM and a *dHash algorithm* to compare the images generated. The TLEA with the TDEM and the dHash algorithm allows the dataset size to grow linearly with the number of epochs.
- We prepare a set of image processing operators, in which we introduce two noise injection operators. The two noise injection operators can effectively improve the anti-interference capability of the model. By introducing these operators, CAugment is able to generate much more diverse image samples that meet for real-world scenarios, which further improves the model robustness and generalization performance.
- We evaluate CAugment on the CIFAR-10/100 [16] and SVHN [29] datasets using the WideResNet [44] and Shake-Shake [7] models. The results show that CAugment outperforms the default approach by up to 17.9% in accuracy on SVHN.

Compared with existing techniques, CAugment is novel in the following respects. First, CAugment employs the novel TLEA and different search heuristics to control the search space, which balances the search efficiency and image diversity. It also provides three-level granularity control to guarantee the quality of generated images, ensuring model accuracy. Second, CAugment combines multiple operators, including two novel noise injector operators, to generate rich and diverse images. Training with noisy images facilitates the resistance of the model to attacks and enhances its robustness. Third, CAugment is the first to combine the TDEM and dHash algorithm to measure the diversity of the augmented image dataset.

2. Background: Search Space

The search space is the set of all possible settings that the search algorithm can explore. It can lead to a manageable search space by simplifying the search algorithm. However, it also reduces the diversity of the image dataset. Therefore, it is crucial to strike a balance between the search space complexity and the ability of finding optimal solutions.

The combination of operators in CAugment has the following characteristics. First, since operators can be used repeatedly, the search space can be infinite. However, the use of the same operator multiple times can be simplified by the fact that combinations of the same operator can be superimposed on each other. Specifically, if a is an operator, x is an image and m_1 and m_2 are the parameter intensities of the operator, we have

$$a(a(x, m_1), m_2) = a(x, m_1 + m_2), \tag{1}$$

resulting in nearly identical effects. This characteristic sets the number of possible combinations of operators at most 16. Additionally, different operators are often not interchangeable, meaning that

$$a_2(a_1(x, m_1), m_2) \neq a_1(a_2(x, m_2), m_1), \tag{2}$$

where a_1 and a_2 are two different operators.

Second, the order of the operators cannot be easily tuned due to the nature of the image processing operators, which consist of three parts, i.e., the geometric transformation, color transformation and noise injection. A geometric transformation alters the distribution of the image without changing its value, and thus a combination of geometric transformation operators has few impacts on the images. On the other hand, color transformation and noise injection can change the images, and changing the order of these operators may cause the loss of image features. Let the average number of parameters of an operator be 1, and the parameter intensity be discretized into 11 levels (only for parameters with continuous intensity values). There exist altogether 16 operators, each of which can be executed at most once. Consequently, the search space grows in a factorial order:

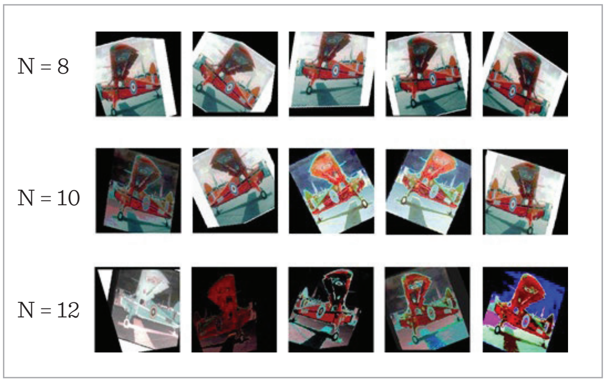
$$S = \sum_{k=1}^{n} k! \times m^{\sum_{i=1}^{k} p} \approx \sum_{k=1}^{n} k! \times m^k$$
$$\approx n! \times m^n = 16! \times 11^{16} \approx 9.6 \times 10^{29}. \tag{3}$$

Here, S represents the search space, n the number of operators, m the discretization degree of the parameters of the operator and p the number of parameters of the operator.

Images generated under different search spaces are significantly different. Due to the factorial expansion of the search space resulting from the addition of op-

Figure 1

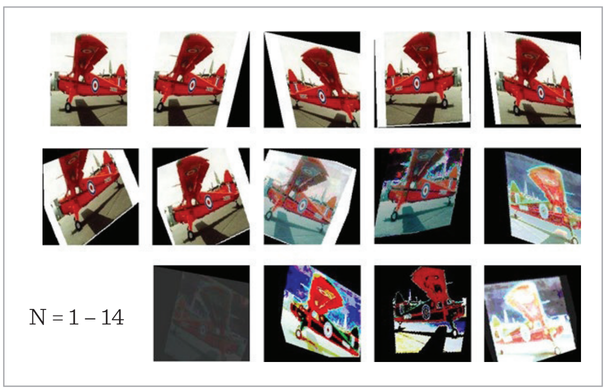
The figure illustrates the images generated by five different combinations when N=8, 10 and 12. Here, N represents the number of operators in the combination. It indicates that there is a significant variation of the generated images among different combinations when the number of operators remains unchanged. This shows it is important to choose a good combination of operators



erators, images generated through this approach do have high diversity and are of potentials for mining useful features. Figure 1 illustrates the images generated through different strategies when using 8, 10 and 12 operators. While some images are readily recognizable, others are more blurred. Thus it is necessary to avoid using poor strategies while ensuring image diversity. It is ineffective to generate diverse images in a simple search space. Figure 2 further shows the impact of augmenting images from N=1 to N=14; an

Figure 2

The images generated by combinations when the number of operators, N, ranges from 1 to 14, arranged from left to right and top to bottom. This indicates that when N is too small, the images suffer from a lack of diversity. However, when N increases, the images become diverse



increases of values of N results in images greatly different from the original image.

3. Approach

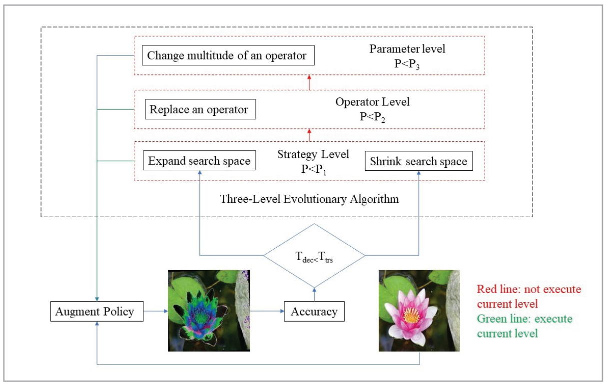
3.1. Overview

CAugment has been designed to explore the potential of the image dataset to enhance the accuracy of model. To achieve this, CAugment adopts an evolutionary algorithm that adjusts the search strategy to meet the training requirements of the model without compromising the efficiency of the algorithm. Additionally, it selects and uses operators, which has a significant impact on the search space and in turn affects the accuracy of the model. To address this, CAugment leverages changes in accuracy as feedback to fine-tune and optimize the selection and use of operators in the algorithm.

The CAugment process is illustrated in Figure 3. We next define the image diversity and introduce two evaluation methods, as well as present the details of the search algorithm.

Figure 3

CAugment and the Three-Level Evolutionary Algorithm



Algorithm 1 draws inspiration from reinforcement learning (RL) [15]. Throughout the training process, it is crucial to observe the fitting performance of the model. Overfitting and underfitting are both undesired. The verification accuracy of the model serves as a reliable metric for observation. Generally, if the verification accuracy continuously decreases, it implies that the model is either overfitting or underfitting. When the model is in an overfitting state, CAugment needs to enhance the diversity of the image set to enable the model to learn new contents. Conversely, if the model is underfitting, it should continue training on the current image set. CAugment employs TLEA to expand the image set. If the verification accuracy of the model continuously decreases, it implies that the model is underfitting. In such cases, CAugment needs to set the limit for the degree of diversity in the image set and allow the model to continue learning from the current set of images. Therefore, CAugment controls the level of diversity in the image set based on the value of the continuously decreasing verification accuracy.

Algorithm 1. CAugment's general process

Require: D: dataset,
 S: strategy,
 T: consecutive times of decline,
 A_p: previous accuracy,
 A: accuracy;
 Initialize S, $T = 0$ and $A_p = 0$;
 for each epoch **do**
  Generate a new dataset $S(D)$;
  Train $S(D)$ and get A;
  if $A_p > A$ **then**
   $T = T + 1$;
  else
   $T = 0$;
  end if
  $A_p = A$
  if $T < 5$ **then**
   Search space tends to expand;
  else
   Search space tends to shrink;
  end if
 end for

CAugment investigates the stability of the algorithm by monitoring the accuracy of the model. As Algorithm 1 shows, CAugment begins with the identity operator as the initial strategy S. At the end of each epoch, CAugment obtains the accuracy A, which is compared with the accuracy A_p of the previous epoch. If the accuracy decreases, TLEA increments T by 1. If the accuracy does not decrease, TLEA resets T 0. If $T <$ 5, S tends to become more complex. Conversely, if the existing strategy is unstable, it needs to be adjusted, even if the search space may be small. We empirically sets the threshold T 5, as when the validation accuracy continuously decreases 5 consecutive times, this is an indication of the poor training performance of the model. In Algorithm 1, this indicates an underfitting of the model. CAugment restricts the search space so that the model can continue learning from the existing image dataset.

CAugment employs an evolutionary algorithm to search for strategies without setting bounds on the search space, resulting in better model accuracy. The evolutionary algorithm controls the strategy complexity through three levels of granularity. The first level controls the operator granularity, the second level controls operator selection and the third level controls the intensity of the operator's parameters. The higher the levels, the more significantly changed the strategy, with the lower execution probability for preventing the algorithm from a fast evolution.

CAugment utilizes the TDEM and dHash algorithm, which will be explained in the next section, to compare the similarity between the images and to verify their ability of generating diverse data. The TDEM evaluates the operator distribution and complexity of strategies. Different batch of data is designed to diversify the generated data to ensure the effectiveness of the data augmentation. Assuming that each data point in the dataset is unique during training, the dHash algorithm compares the changes in the same picture under different strategies in order to confirm the increase of data.

3.2. Three-level Evolutionary Algorithm

The TLEA is inspired by genetic algorithms (GAs) [25] and TrivialAugment (TA) [26]. TA shows that data augmentation using a single random operator has some effect. However, the images generated by a single operator are not diverse. It is necessary to increasing the number of operators for generating diverse images. Algorithm 2 uses multiple operators for data augmentation, but CAugment cannot simply select multiple operators from the operator set and combine them, as poor combinations of operators may cause important features in the images missing. CAugment employs the TLEA to allow the combination to evolve based on the current training status. At the end of each epoch, there is a probability of either retaining the existing operator combination or changing the combination. Unlike traditional GAs, TLEA introduces a hierarchical approach to control the genetic process. Since the number of operators greatly affects the quality of the combinations, TLEA ensures the images resulting from the combinations to be suitable for training. Through different levels of evolution, CAugment can control the magnitude of changes in the combination, thereby preventing it from becoming worse and worse.

CAugment adopts the TLEA that utilizes three levels of operations to adjust the degree of image diversity. These levels are organized in decreasing order of granularity:

- Level 1: Strategy level. This level adds or removes an operator, affecting the size of the search space. The impact of adding or removing operators on the search space becomes significant as the number of combined operators increases.
- Level 2: Operator level. This level replaces an operator to explore the search space. Replacing an operator has an effect on the strategy. However, it does not change the size of the search space.
- Level 3: Parameter level. This level adjusts the intensity of an operator's parameter, which has little impact on the overall strategy but is useful for fine-tuning.

Algorithm 2. Three-Level Evolutionary Algorithm

```
Require:  S: strategy,
  P̄: mean probability,
  P: random sampling probability,
  P1: strategy level probability,
  P2: operator level probability,
  P3: parameter level probability,
  Flag: determine the trend of image changes;
  m: parameter intensity
  Initialize P̄ = Σ_{n=1}^{3} P
  if P̄ < P1 then
    if Flag == 1 then
      S add one unselected operator
    else
      S reduce one operator
    end if
  else if P̄ < P2 then
    Replace one operator in S
  else if P̄ < P3 then
    Replace m of an operator in S
  end if
```

The general flow of the TLEA is shown in Algorithm 2. The probabilities of the three levels are set differently since they have different effects on the strategy. These probabilities increase iteratively. The three-level operations are executed step by step. When the upper level is not executed, the lower level is executed with a certain probability. Finally, CAugment strictly controls whether to increase or decrease the number of operators at the level 1, which provides a stable strategy.

The TLEA aims to continuously adjust the search strategy, which is consistent with the changing demands on images during model training. The TLEA explores the potential of the data by expanding the search space, allowing the model to learn from a diverse range of images and preventing from overfitting. By gradually increasing the degree of change in the image, the training model learns more diverse features, thereby improving its generalization ability. However, as the model is trained on more data, the risk of underfitting tends to increase. When the verification accuracy of the model continuously decreases 5 times or more, the TLEA fixes the search space. The algorithm is shown in Algorithm 2.

TLEA is novel in the following aspects: a) Efficiency: It can flexibly control the search space and support efficient search. b) Diverse images: It supports a combination of multiple operators to generate a diverse set of images. c) Image quality: It provides three levels of granularity control to guarantee the quality of the generated images and ensure model accuracy.

Efficiency: Image data augmentation techniques typically employ two types of methods. The first applies simple transformations to images using image operators, while the second generates images using GAN models. For the first type, each operator implementation is simple and efficient. However, each operator includes a set of parameters, and different operators can be combined, which means that selecting a strategy composed of multiple operators is not easy. For the second type, we only need to design the model architecture. Generally, this type of methods produce images of reasonable quality, but requires much time for training the model. We choose the first type of methods to maintain algorithm efficiency.

Control image diversity: Another core issue is the generation of strategies. Due to the high efficiency of the image operators, it hardly affects the performance by combining operators, because some technologies achieve data augmentation by manually setting operators and parameters. However, it usually requires specialized expertise for setting parameters manually and multiple experiments. Moreover, the strategy set by this method often lacks portability.

We utilize the TLEA to adaptively adjust strategies during training based on the validation accuracy of the model, aiming to generate the desired images for the model. The TLEA also guarantees the test accuracy of the model. It is worth noting that due to the increasingly complex during evolution, there is a risk of underfitting the model. When the validation accuracy of the model continuously decreases, the TLEA constrains the search space. The strategy evolves to combine multiple operators, which determines the generated images. The number of operators determines the search space since the more operators, the more variable parameters, the larger the search space. When the validation accuracy of the model continuously decreases, the strategy removes an operator to shrink the search space.

4. Image Diversity

Image diversity means that the new images generated from the original image through various strategies have differences from each other. CAugment defines image diversity to describe the potential of an image for training a model. These generated images ensure that the model is exposed to a greater variety of images, allowing the model to be trained more sufficiently.

4.1. Three-Dimensional Evaluation Method

CAugment employs the TDEM to quantify the image diversity based on the changes made by the operators. The operators used in the algorithm are classified into three categories, namely, *geometric transformation* operators, *color transformation* operators and *noise injection* operators. Images generated by operators of different categories are different from each other, whereas those generated by operators of the same category tend to be more similar. The diversity of the images is thus quantified along with three dimensions. To further quantify the dimension, we discretize those parameters whose values are continuous. For boolean parameters, we set its values 0 and 1. The formula for calculating the quantized value in each dimension is:

$$Q = \sum_{i=1}^{n} m_i, \tag{4}$$

where Q represents the total quantized value of the dimension, n the number of operators included in that dimension, and m_i the strength of the operator's parameters. The degree of changes in three dimensions is calculated by summing the parameter intensity values of each type of operator. Our operators include 5 *geometric transformation* operators, 8 *color transformation* operators, 2 *noise injection* operators and 1 operator that does not modify images in any way.

Table 1
A description of each operator

Identity	Do nothing with the image.
ShearX(Y)	Shear the image along the horizontal (vertical) axis with rate magnitude.
TranslateX(Y)	Translate the image in the horizontal (vertical) direction by magnitude number of pixels.
Rotate	Rotate the image magnitude degrees.
AutoContrast	Maximize the the image contrast, by making the darkest pixel black and lightest pixel white.
Equalize	Equalize the image histogram.
Solarize	Invert all pixels above a threshold value of magnitude.
Posterize	Reduce the number of bits for each pixel to magnitude bits.
Contrast	Control the contrast of the image.
Color	Adjust the color balance of the image, in a manner similar to the controls on a colour TV set.
Brightness	Adjust the brightness of the image.
Sharpness	Adjust the sharpness of the image.
SaltandPepper Noise	Add salt and pepper noise to the image.
Gaussian Noise	Add gaussian noise to the image.

TDEM is novel in (1) validating the increase of the number of images and (2) measuring the quality of the images. In addition, TDEM is closely related to the image operator library. Existing techniques typically use geometric transformation operators and color transformation operators. We further introduce noise injection operators. We classify the operators into three categories and evaluate the diversity of the images from three dimensions. The results show that CAugment has indeed diversified the existing image set. Furthermore, we compare the results using the dHash algorithm and draw out the same conclusion. TDEM is also helpful for studying the relationship between operators and image sets. It clearly shows that the generated images are different from the original images.

4.2. dHash Algorithm

CAugment also utilizes the dHash algorithm to measure the similarity between images. There are three commonly used methods for image similarity: average hashing (aHash) [10], perceptual hashing (pHash) [32], and differential hashing (dHash). The aHash algorithm is highly efficient but less accurate, while the pHash algorithm is more accurate but less efficient. Comparatively, the dHash algorithm uses gradient information and can achieve accuracy similar to pHash while maintaining the efficiency of aHash.

The dHash algorithm is calculated in the following steps:

1 Resizing the image: The image is scaled down to a size of 9x8, resulting in 72 pixels. This step discards any image differences caused by varying sizes and ratios.

2 Simplifying the color: The image color is simplified to 64-level grayscale.

3 Calculating the difference value: For each row of the matrix, the dHash algorithm calculates the difference between two adjacent pixels (the left pixel minus the right pixel) to obtain eight different difference values. This step produces a total of 64 difference values.

4 Processing the difference value: If the difference value is greater than or equal to 0, the result is recorded as 1; otherwise, it is set 0.

5 Obtaining the hash value: The dHash algorithm combines the 64 resulting values to obtain a hash value, which serves as the unique "fingerprint" of the image. The combination order of the 64 values of each image must be consistent.

6 Comparing hash values: The dHash algorithm compares the hash values of two images bit by bit to determine whether the two images are the same.

5. Experiment

- RQ1. Is CAugment competitive enough compared with the baselines?
- RQ2. Does CAugment perform well on a reduced dataset?
- RQ3. Are the images generated by CAugment diverse enough?

5.1. Setup and Design

We conduct experiments on three benchmarks: CIFAR-10/100 and SVHN. The baselines used in this experiment include AutoAugment (AA) [3], Population Based Augmentation (PBA) [11], Fast AutoAugment (FastAA) [21], RandAugment (RA) [4], TA, Deep AutoAugment (DeepAA) [46] and TeachAugment [39]. We provide 16 image processing operators, namely, Identity, ShearX, ShearY, TranslateX, TranslateY, Rotate, AutoContrast, Equalize, Solarize, Posterize, Contrast, Color, Brightness, Sharpness, SaltandPepper Noise and Gaussian Noise. Table 1 shows the descriptions of these operators.

The experimental metrics include accuracy and image diversity. We utilize CAugment to diversify the dataset for model training and evaluate its effectiveness across different datasets and models. The only parameter that needs to be set is the triggering probability for the three levels in TLEA. We set them as P_1=0.2, P_2=0.4 and P_3=0.8. In addition to generating diverse images, training a good model requires the proper setting of hyperparameters. However, CAugment does not focus on configuration of the hyperparameters. We use the hyperparameter configuration files of TA to facilitate better comparative experiments. The evaluation of image diversity includes the TDEM and dHash algorithm.

The CPU used in this study is an i7-13700K processor, while the GPU is a 3090 graphics card. We only evaluate the performance of small datasets and small models in this study.

5.2. Image Diversity Assessment

We use the TDEM and dHash algorithm to evaluate the image diversity. The dHash algorithm focuses on the contrast between two images. It compares images generated according to strategies for the same original image in adjacent epochs. The changes in the strategy lead to variations in the generated images. Similarity of the images is achieved by comparing two consecutive epochs. We extract 10% of the dataset for evaluation. The results are presented in Table 2, where e represents epochs.

Eventually, the amount of data grows linearly, following the Equation $y=kx$, where y is the total amount of

Table 2
Evaluation of image diversity by the dHash algorithm

dHash	CIFAR-10	CIFAR-100	SVHN
Default	1/e	1/e	1/e
AA	2/e	2/e	2/e
RA	2/e	2/e	2/e
Ours	0.893	0.884	0.797

data for model training, x is the original dataset and k is the growth rate of the data. While the value of k for the default method is $1/e$ since it does not perform data augmentation. Due to the use of fixed strategies in AA and RA, they only increase the data by a factor of two. The value of k tends to be 0.1 to 0.2 lower than the ideal value of 1.

There are a few reasons for this. First, if the strategy remains unchanged, the amount of data does not increase. Second, the initial strategy preserves the original images, which helps the model learn the features of the original dataset more effectively. Finally, since changing the strategy may result in only slight changes to the image, we consider the image before and after the change as the same image. All of these factors lead to the lower growth rate of our data.

The TDEM quantifies the complexity of the dataset transformations, which reflects the superposition of the number and intensity of operators. The metric used after model training is the degree of 3D transformation of the dataset. For evaluation, we use the WRN-28-2 model. The results are presented in Table 3.

Table 3
Evaluation of Image Diversity by TDEM

<table>
<tr><th colspan="2">TDEM</th><th>CIFAR-10</th><th>CIFAR-100</th><th>SVHN</th></tr>
<tr><td rowspan="3">AA</td><td>Geometry (5)</td><td colspan="2">1.75</td><td>2.36</td></tr>
<tr><td>Color (9)</td><td colspan="2">3.72</td><td>1.78</td></tr>
<tr><td>Cutout and Sample Pairing</td><td colspan="2">0.00</td><td>0.00</td></tr>
<tr><td rowspan="3">Ours</td><td>Geometry (5)</td><td>2.4</td><td>0.5</td><td>0.5</td></tr>
<tr><td>Color (8)</td><td>3.6</td><td>2.8</td><td>2.8</td></tr>
<tr><td>Noise (2)</td><td>0.4</td><td>0.4</td><td>0.4</td></tr>
</table>

The results show the effectiveness of different operators in generating diverse images. It indicates that while noise injection remains useful, it should be used judiciously and not excessively. In contrast, color operators are more effective than geometric operators, indicating that they are the most suitable for the three datasets under study. With training data from CIFAR-10, our CAugment approach is able to rapidly diversify images using various operators. In comparison, the operator distribution of AA is consistent with our results but its geometry operator is more effective than the color operator for the SVHN dataset.

5.3. Performance

Table 4 presents the CAugment experimental results and the baselines.

- **CIFAR-10:** CAugment achieves the best performance on the WRN-28-10, WRN-40-2 and SS (26 2×32d) models. Specifically, CAugment outperforms the best baseline by 0.3% in terms of test accuracy on the WRN-40-2 model. For the remaining models, CAugment achieves performances comparable to the state-of-the-art.
- **CIFAR-100:** CAugment attains the highest performance on the WRN-28-10 model. Furthermore, for the other models, CAugment shows performance on par with the state-of-the-art.
- **SVHN:** CAugment improves accuracy by 0.05% over the best baseline on WRN-28-10 and achieves a significant 17.9% improvement compared to the default model. These results show the competitiveness of CAugment.

CAugment equipped with TLEA for automated data augmentation is competitive with existing baselines. Furthermore, the time taken by CAugment to transform the image dataset is negligible. While maintaining algorithm efficiency and high model testing accuracy, CAugment also enhances the diversity of images, which reduces the complexity of data preprocessing for the users and allows the model to learn more valuable features from a wide range of images.

It is worth noting that it does not outperform the best baseline in some aspects. One limitation is that CAugment does not achieve optimal results in training hyperparameters. It is necessary to adjust the hyperparameters according to the specific CAugment characteristics. Another limitation is that it takes some effort for the

Table 4
Comparison of CAugment and the baselines on different datasets and models

	Default	AA	PBA	FastAA	RA	TA	DeepAA	TeachAugment	Ours
CIFAR-10									
WRN-28-10	96.13	97.34	97.42	97.3	97.3	97.46	97.56	97.5	**97.67**
WRN-40-2	94.7	96.3	-	96.3	-	96.32	-	-	**96.63**
SS(26 2×32d)	96.45	97.53	97.46	97.5	-	-	-	-	**97.64**
SS(26 2×96d)	97.14	98.01	97.97	98	98	**98.21**	98.11	98	98.06
SS(26 2×112d)	97.18	**98.11**	*97.97*	*98.1*	-	-	-	-	98.1
CIFAR-100									
WRN-28-10	81.2	82.91	83.27	82.8	83.3	84.33	84.02	83.2	**84.46**
WRN-40-2	74	79.3	-	79.4	-	**79.86**	-	-	79.81
SS (26 2×96d)	82.95	85.72	84.69	85.4	-	**86.19**	85.19	85.5	85.97
SVHN									
WRN-28-10	81.16	85.87	98.82	98.9	99	98.9	-	-	**99.05**

model to be trained with new data. Therefore, it is beneficial to retain the strategy for a certain period after making changes, even if it may involve certain trade-offs, such as compromising image diversity. Nonetheless, the results show that CAugment is a promising approach for automatic data augmentation. Future work can focus on addressing these limitations and further enhancing the performance of CAugment.

5.4. Reduced Dataset

We further evaluate the impact of CAugment on accuracy by reducing the size of the dataset. We find that when the dataset has sufficient data, CAugment is highly effective. This is mainly due to the architecture of the model. However, when the dataset is reduced in size, CAugment can compensate for the shortcomings of the reduced data by further exploring the images and maximizing the potential of images throughout the entire training process.

We use the SVHN (core) dataset to observe the impact of CAugment on model accuracy and compared it with the baselines. Table 5 shows the accuracy obtained by training the models with each method on the SVHN (core) dataset. The results show that training the model with CAugment on the reduced dataset achieves approximately 1% higher accuracy compared to the default method. It shows that CAugment is competitive.

Table 5
Effects of CAugment on the SVHN (core) dataset

	Default	AA	RA	TA	Ours
WRN-28-2	96.7	98.0	**98.3**	-	97.67
WRN-28-10	96.9	98.1	**98.3**	98.11	98.01

As the amount of training data decreases, the generated data may not be fully trained, leading to the accuracy of the model not exceeding that of the best baseline. In this case, it may be beneficial for prolonging the duration of a certain strategy for achieving test accuracy.

6. Related Work

Data augmentation is a technique used to maximize the potential of existing datasets, or generate new data with diverse features to facilitate sufficient learning of models. There are two main types of data augmentation: offline and online. *Offline augmentation* transforms data before the model training process, which results in a linear increase in the memory occupied by the data. *Online augmentation* transforms data after reading the batch, enabling an increase in data without consuming additional storage space. Online augmentation is currently the mainstream. In the field of CV, various data augmentation methods have been proposed and summarized in [27].

AA was an early study on automatic data augmentation. Its core idea is to learn a data augmentation strategy and apply it to augment the data. The final training effect is better than that of existing data augmentation techniques. First, it defines a search space mainly composed of three dimensions, which are operator selection, operator execution probability and parameter intensity. The search algorithm employs RL and uses the accuracy of the model as a reward signal to continuously adjust the strategy. While AA points out a new direction for data augmentation techniques, it is time-consuming and impractical. Hence, subsequent studies focused on reducing the search time.

Two additional approaches, PBA and FastAA, focus on reducing the time required to search for strategies. PBA utilizes a two-step process that involves exploiting and exploring. During the exploitation step, PBA generates clones of model parameters with high performance. During the exploring step, the existing parameters are resampled or randomly perturbed to obtain the required data. FastAA divides the dataset into K folds, and each fold of the dataset is divided into two parts, D_A and D_M. D_M is used for submodel training. Unlike AA, the probability and parameter intensity of each operator in PBA are continuous. The search algorithm uses Bayesian optimization. It selects the top-N search strategies and combines the strategies obtained from the K-fold dataset to obtain the final strategy. Both PBA and FastAA have the ability to reduce the search time for data augmentation while achieving an accuracy rate similar to that of AA. They make automatic data augmentation more practical. DADA [20] employs a differentiable optimization approach to obtain the final strategy in a continuous search space. Adversarial data augmentation is another approach in Adversarial AutoAugment (Adv. AA) [45], while [47] generates strategies by extracting image features.

RA proposes a simplified search space by removing the probability of operators and only retaining the number and parameter intensity of operators. This simplified space is encapsulated into a function. RA also investigates the effect of parameter intensity and model accuracy under different models and dataset sizes, finding that the optimal parameter intensity of an operator varies with the model and dataset. RA greatly reduces the search time and achieves similar effects to AA. On the other hand, [23] introduces the concept of diversity in data augmentation and establishes a connection between diversity and regularization to improve the regularization effect of the model.

TA simplifies RA further by randomly selecting an operator and parameter intensity for each image. TA significantly improves the accuracy of the model and emphasizes the importance of data augmentation at the image level. Smart (Sampling) Augment [28] improves the algorithm of both RA and TA and applies them to the field of image segmentation.

TeachAugment leverages the Teacher model to produce informative images, enabling more effective model training through the generation of diverse and informative images. DeepAA progressively constructs a multi-layer data augmentation pipeline from scratch, adding one augmentation layer at a time until the model converges. DeepAA significantly improves the accuracy of the model while maintaining low computation cost.

The field of image processing has a broad range of research directions. One area of study is defect detection, as explored by [22]. In the medical field, image processing has been utilized for a variety of purposes, such as in [43, 6, 31, 9]. Climate detection and management is another area where image processing has been applied such as [24]. Filters used for image restoration, image augmentation and denoising have also been developed by researchers such as [18]. [5] uses sensor images for real-time strain prediction. Material hardness detection has been investigated using image processing techniques such as [35]. [13] applies image processing to carbon nanotubes. [37, 2] explore the use of data augmentation in image segmentation. Finally, image processing has also been applied in identification tasks such as [41].

Data augmentation techniques have also been applied in the NLP field. [38, 30, 1, 42] have used data augmentation to synthesize data and address the issue of data scarcity. Additionally, [34, 19, 33] have explored various data augmentation techniques used in the field of NLP and evaluated their effectiveness. Moreover, data augmentation techniques were utilized in [20] to enhance the robustness of NLP models. Furthermore, [36] employs ChatGPT to generate new data, showcasing another innovative application of data augmentation techniques in NLP.

7. Conclusion

In this paper, we present CAugment, a novel approach to diversify dataset by combining various image processing operations. Our experiments on the SVHN dataset shows that CAugment outperforms the default approach by 17.9% in accuracy, which shows its competitiveness. We also evaluates the performance of CAugment on reduced datasets and find that it remains competitive in such scenarios. These results indicate that diversifying images through CAugment is an effective approach, which should be further investigated. Furthermore, we evaluate the diversity of the images generated by CAugment. The results show that the number of different images increases linearly with the number of epochs. It clearly indicates that CAugment improves the diversity of the image datasets. Moreover, the results also show that the diversity is further enhanced by the operators used in our study.

Data Sharing Agreement

The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, author-ship, and/or publication of this article.

Funding

The author(s) received no financial support for the research.

References

1. Bayer, M., Kaufhold, M. A., Buchhold, B., Keller, M., Dallmeyer, J., Reuter, C. Data Augmentation in Natural Language Processing: A Novel Text Generation Approach for Long and Short Text Classifiers. International Journal of Machine Learning and Cybernetics, 2022, 14(1), 135-150. https://doi.org/10.1007/s13042-022-01553-3
2. Bhosale, S., Krishna, A., Wang, G., Mueller, K. Improving CT Image Segmentation Accuracy Using Stylegan Driven Data Augmentation. Computing Research Repository, 2023.
3. Cubuk, E. D., Zoph, B., Mane, D., Vasudevan, V., Le, Q. V. Autoaugment: Learning Augmentation Policies from Data. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Computing Research Repository, 2018, abs/1805.09501. https://doi.org/10.1109/CVPR.2019.00020
4. Cubuk, E. D., Zoph, B., Shlens, J., Le, Q. V. Randaugment: Practical Automated Data Augmentation with a Reduced Search Space. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2020, 702-703. https://doi.org/10.1109/CVPRW50498.2020.00359
5. de Castro, L. D., Scabini, L., Ribas, L. C., Bruno, O. M., Oliveira Jr, O. N. Machine Learning and Image Processing to Monitor Strain and Tensile Forces with Mechanochromic Sensors. Expert Systems with Applications, 2023, 212, 118792. https://doi.org/10.1016/j.eswa.2022.118792
6. Doshi, R., Hiran, K. K., Doppala, B. P., Vyas, A. K. Deep Belief Network-Based Image Processing for Local Directional Segmentation in Brain Tumor Detection. Journal of Electronic Imaging, 2023, 32(6). https://doi.org/10.1117/1.JEI.32.6.062502
7. Gastaldi, X. Shake-Shake Regularization. Computing Research Repository, 2017, abs/1705.07485.
8. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y. Generative Adversarial Networks. Communications of the ACM, 2020, 63(11), 139-144. https://doi.org/10.1145/3422622
9. Gubert, P. H., Costa, M. H., Silva, C. D., Trofino-Neto, A. The Performance Impact of Data Augmentation in CSP-Based Motor-Imagery Systems for BCI Applications. Biomedical Signal Processing and Control, 2020, 62, 102152. https://doi.org/10.1016/j.bspc.2020.102152
10. Haviana, S. F. C., Kurniadi, D. Average Hashing for Perceptual Image Similarity in Mobile Phone Application. Journal of Telematics and Informatics, 2016, 4(1), 12-18.
11. Ho, D., Liang, E., Chen, X., Stoica, I., Abbeel, P. Population-Based Augmentation: Efficient Learning of Augmentation Policy Schedules. Proceedings of Machine Learning Research, 2019, 97, 2731-2741.
12. Hochreiter, S., Schmidhuber, J. Long Short-Term Memory. Neural Computation, 1997, 9(8), 1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735

13. Imtiaz, T., Doumani, J., Tay, F., Komatsu, N., Marcon, S., Nakamura, M., Ghosh, S., Baydin, A., Kono, J., Zubair, A. Facile Alignment Estimation in Carbon Nanotube Films Using Image Processing. Signal Processing, 2023, 202, 108751. https://doi.org/10.1016/j.sigpro.2022.108751
14. Jing, Y., Yang, Y., Feng, Z., Ye, J., Yu, Y., Song, M. Neural Style Transfer: A Review. IEEE Transactions on Visualization and Computer Graphics, 2019, 26(11), 3365-3385. https://doi.org/10.1109/TVCG.2019.2921336
15. Kaelbling, L. P., Littman, M. L., Moore, A. W. Reinforcement Learning: A Survey. Journal of Artificial Intelligence Research, 1996, 4, 237-285. https://doi.org/10.1613/jair.301
16. Krizhevsky, A., Hinton, G. Learning Multiple Layers of Features from Tiny Images. 2009.
17. Kshirsagar, S., Falk, T. H. Cross-Language Speech Emotion Recognition Using Bag-of-Word Representations, Domain Adaptation, and Data Augmentation. Sensors, 2022, 22(17), 6445. https://doi.org/10.3390/s22176445
18. Kumar, G. K., Akurati, R. R., Reddy, V. H. P., Cheemalakonda, S., Chagarlamudi, S., Dasari, B., Shaik, S. S. Area-, Power-, and Delay-Optimized 2D FIR Filter Architecture for Image Processing Applications. Circuits, Systems, and Signal Processing, 2022, 42(2):780-800, 2023. https://doi.org/10.1007/s00034-022-02232-y
19. Li, B., Hou, Y., Che, W. Data Augmentation Approaches in Natural Language Processing: A Survey. AI Open, 2022, 3, 71-90. https://doi.org/10.1016/j.aiopen.2022.03.001
20. Li, Y., Hu, G., Wang, Y., Hospedales, T., Robertson, N. M., Yang, Y. DADA: Differentiable Automatic Data Augmentation. Computing Research Repository, 2020, abs/2003.03780.
21. Lim, S., Kim, I., Kim, T., Kim, C., Kim, S. Fast AutoAugment. Advances in Neural Information Processing Systems, 2019, 32, 6665-6675.
22. Ling, Q., Isa, N. A. M. Printed Circuit Board Defect Detection Methods Based on Image Processing, Machine Learning, and Deep Learning: A Survey. IEEE Access, 2023, 11, 15921-15944. https://doi.org/10.1109/ACCESS.2023.3245093
23. Liu, Z., Jin, H., Wang, T. H., Zhou, K., Hu, X. DivAug: Plug-in Automated Data Augmentation with Explicit Diversity Maximization. 2021 IEEE/CVF International Conference on Computer Vision (ICCV), 2021, 4762-4770. https://doi.org/10.1109/ICCV48922.2021.00472
24. Mansour, R. F., Alabdulkreem, E. Disaster Monitoring of Satellite Image Processing Using Progressive Image Classification. Computer Systems Science and Engineering, 2023, 44(2), 1161-1169. https://doi.org/10.32604/csse.2023.023307
25. Mirjalili, S., Mirjalili, S. Genetic Algorithm. Evolutionary Algorithms and Neural Networks: Theory and Applications, 2019, 780, 43-55. https://doi.org/10.1007/978-3-319-93025-1_4
26. Müller, S. G., Hutter, F. TrivialAugment: Tuning-Free Yet State-of-the-Art Data Augmentation. 2021 IEEE/CVF International Conference on Computer Vision (ICCV), 2021, 774-782. https://doi.org/10.1109/ICCV48922.2021.00081
27. Mumuni, A., Mumuni, F. Data Augmentation: A Comprehensive Survey of Modern Approaches. Array, 2022, 16, 100258. https://doi.org/10.1016/j.array.2022.100258
28. Negassi, M., Wagner, D., Reiterer, A. Smart (Sampling) Augment: Optimal and Efficient Data Augmentation for Semantic Segmentation. Algorithms, 2022, 15(5), 165. https://doi.org/10.3390/a15050165
29. Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., Ng, A. Y. Reading Digits in Natural Images with Unsupervised Feature Learning. 2011.
30. Nguyen, D. T., Tran, T. Natural Language Generation from Universal Dependencies Using Data Augmentation and Pre-Trained Language Models. International Journal of Intelligent Information and Database Systems, 2023, 16(1), 89-105. https://doi.org/10.1504/IJIIDS.2023.128292
31. Nguyen, K. P., Fatt, C. C., Treacher, A., Mellema, C., Trivedi, M. H., Montillo, A. Anatomically Informed Data Augmentation for Functional MRI with Applications to Deep Learning. Medical Imaging 2020: Image Processing, 2020, 11313, 172-177. https://doi.org/10.1117/12.2548630
32. Niu, X. M., Jiao, Y. H. An Overview of Perceptual Hashing. Acta Electronica Sinica, 2008, 36(7), 1405.
33. Okimura, I., Reid, M., Kawano, M., Matsuo, Y. On the Impact of Data Augmentation on Downstream Performance in Natural Language Processing. Proceedings of the Third Workshop on Insights from Negative Results in NLP, 2022, 88-93. https://doi.org/10.18653/v1/2022.insights-1.12
34. Pellicer, L. F. A. O., Ferreira, T. M., Costa, A. H. R. Data Augmentation Techniques in Natural Language Processing. Applied Soft Computing, 2023, 132, 109803. https://doi.org/10.1016/j.asoc.2022.109803
35. Polanco, J. D., Jacanamejoy-Jamioy, C., Mambuscay, C. L., Piamba, J. F., Forero, M. G. Automatic Method for Vickers Hardness Estimation by Image Pro-

cessing. Journal of Imaging, 2022, 9(1), 8. https://doi.org/10.3390/jimaging9010008

36. Sahu, G., Rodriguez, P., Laradji, I. H., Atighehchian, P., Vazquez, D., Bahdanau, D. Data Augmentation for Intent Classification with Off-the-Shelf Large Language Models. Proceedings of the 4th Workshop on NLP for Conversational AI, 2022, 47-57. https://doi.org/10.18653/v1/2022.nlp4convai-1.5

37. Sakalik, P., Hudec, L., Jakab, M., Benešová, V. and Fabian, O. Synthesis for Dataset Augmentation of H&E Stained Images with Semantic Segmentation Masks. Proceedings of the 18th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, 2023, 4, 873-880. https://doi.org/10.5220/0011679300003417

38. Sujana, Y., Kao, H. Y. LiDA: Language-Independent Data Augmentation for Text Classification. IEEE Access, 2023, 11, 10894-10901. https://doi.org/10.1109/ACCESS.2023.3234019

39. Suzuki, T. Teachaugment: Data Augmentation Optimization Using Teacher Knowledge. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, 10904-10914. https://doi.org/10.1109/CVPR52688.2022.01063

40. Wang, D. Z., Liang, J. Y. Research and Design of Theme Image Crawler Based on Difference Hash Algorithm. IOP Conference Series: Materials Science and Engineering, 2019, 563(4), 042080. https://doi.org/10.1088/1757-899X/563/4/042080

41. Wu, Q., Dai, P., Chen, P., Huang, Y. Deep Adversarial Data Augmentation with Attribute Guided for Person Re-Identification. Signal, Image and Video Processing, 2019, 15(4), 655-662. https://doi.org/10.1007/s11760-019-01523-3

42. Xu, F., Dan, Y., Yan, K., Ma, Y., Wang, M. Low-Resource Language Discrimination Toward Chinese Dialects with Transfer Learning and Data Augmentation. ACM Transactions on Asian and Low-Resource Language Information Processing, 2021, 21(2), 1-21. https://doi.org/10.1145/3473499

43. Yuan, D., Liu, Y., Xu, Z., Zhan, Y., Chen, J., Lukasiewicz, T. P. Painless and Accurate Medical Image Analysis Using Deep Reinforcement Learning with Task-Oriented Homogenized Automatic Pre-Processing. Computers in Biology and Medicine, 2023, 153, 106487. https://doi.org/10.1016/j.compbiomed.2022.106487

44. Zagoruyko, S., Komodakis, N. Wide Residual Networks. Procedings of the British Machine Vision Conference 2016, Computing Research Repository, 2016, abs/1605.07146. https://doi.org/10.5244/C.30.87

45. Zhang, X., Wang, Q., Zhang, J., Zhong, Z. Adversarial Autoaugment. Computing Research Repository, 2019, abs/1912.11188.

46. Zheng, Y., Zhang, Z., Yan, S., Zhang, M. Deep Autoaugment. Computing Research Repository, 2022, abs/2203.06172.

47. Zhou, F., Li, J., Xie, C., Chen, F., Hong, L., Sun, R., Li, Z. Metaaugment: Sample-Aware Data Augmentation Policy Learning. Proceedings of the AAAI Conference on Artificial Intelligence, 2021, 35(12), 11097-11105. https://doi.org/10.1609/aaai.v35i12.17324