IDA: Improved Data Augmentation Applied to Salient Object Detection

Daniel V. Ruiz and Bruno A. Krinski and Eduardo Todt
Department of Informatics, Federal University of Parana, Brazil
Email: {dvruiz, bakrinski, todt}@inf.ufpr.br

***Abstract*—In this paper, we present an Improved Data Augmentation (IDA) technique focused on Salient Object Detection (SOD). Standard data augmentation techniques proposed in the literature, such as image cropping, rotation, flipping, and resizing, only generate variations of the existing examples, providing a limited generalization. Our method combines image inpainting, affine transformations, and the linear combination of different generated background images with salient objects extracted from labeled data. Our proposed technique enables more precise control of the object's position and size while preserving background information. The background choice is based on an inter-image optimization, while object size follows a uniform random distribution within a specified interval, and the object position is intra-image optimal. We show that our method improves the segmentation quality when used for training state-of-the-art neural networks on several famous datasets of the SOD field. Combining our method with others surpasses traditional techniques such as horizontal-flip in 0.52% for F-measure and 1.19% for Precision. We also provide an evaluation in 7 different SOD datasets, with 9 distinct evaluation metrics and an average ranking of the evaluated methods.**

I. Introduction

A visual scene is a complex structure composed of many different regions and objects, with a large variety of color, size, and texture. The human visual system has the natural ability to filter such complex structures and focus on the most attended regions, also called salient regions, making it faster for our brain to analyze and understand all regions and objects located in the scene. Aiming to replicate this ability in machines the Salient Object Detection (SOD) research field focuses on studying Computer Vision techniques to find the most salient objects in images [1].

In recent decades, the SOD literature presented an impressive growth in the number of novel and promising approaches. Recent works, which are based on Deep Learning techniques, have shown remarkable results in the field [2], [3]. Due to its high precision and generalization abilities, Deep Learning-based methods can find the salient regions of images with higher reliability. However, such methods weak point is the amount of data required to train them. For that reason, popular datasets such as ImageNet [4] and PLACES [5] are composed of millions of images.

Moreover, besides obtaining a significant number of images to compose a dataset in a supervised learning approach, the images have to be labeled. The labeling of images is specialized and laborious work and can be even more expensive in segmentation tasks like SOD, which requires a pixel-wise segmentation mask for each image in the dataset.

In general, two approaches are usually applied to bypass such a necessity of data: transfer learning and data augmentation. The former makes use of pre-trained models in more massive datasets, like ImageNet. In this strategy, the objective is to use generic features already learned by the model and then fine-tuning it to the proposed problem with small datasets [6]. The latter makes use of different augmentation techniques like horizontal/vertical flip, image cropping, rotation, and others, to synthetically generate new training samples based on the existing ones [7]. While others deal with domain adaptation to expand the data available in novelty ways as in [8]. In this work, we address the data problem by proposing a technique called Improved Data Augmentation (IDA).

The SOD problem is addressed in the literature as a cross-testing problem [9], with the training being performed in one dataset and the testing being performed in other datasets, utterly different from the training dataset. When analyzing the SOD datasets proposed in the literature, we noticed that the training dataset is not generic enough. In general, the training dataset contains certain biases such as large salient objects positioned in the center of the image, with size and position distribution very different from the testing datasets. Thus, our primary goal is to generate new salient examples in compliance with a more widespread distribution that can encompass the testing datasets' distribution to train more precise models and prevent overfitting to the original training dataset.

In this paper, we propose an improved approach to data augmentation in the context of SOD. Our method uses a linear combination of two different images. The resulting image contains in the foreground: a salient object segmented from its original background subsequently affinely transformed, and a full background generated created using image inpainting to erase its labeled objects. Our proposed method can be summarized in five steps: background generation, feature extraction, distance optimization, affine transformation, and linear combination. The implementation of the method is publicly available [1].

To demonstrate the effectiveness of this technique, we present quantitative and qualitative results obtained by augmenting the public available dataset DUTS [10] to train the state-of-the-art deep neural network PoolNet [11] with the

[1] https://github.com/VRI-UFPR/IDA

Res2Net (Res2Net) backbone [12] for Salient Object Detection (SOD).

II. Related Work

The majority of data augmentations techniques proposed in the SOD literature are restricted to affine transformations, random crop, and random flip. Perazzi *et al.* [13] proposed affine transformations to generate new images by scaling and translating the training images. Guo *et al.* [14] and Liu *et al.* [11] used random flip as data augmentation. Wei *et al.* [15], and Wang *et al.* [16] utilized image resize, random crop and random horizontal flip.

To the best of our knowledge, Ruiz *et al.* [17] proposed the first method incorporating an inpainting technique for data augmentation of salient objects images. However, this method, named ANDA, does not guarantee intra-image optimal salience, since object placement simply follows a random uniform distribution.

Chen et al. [18] proposed a data augmentation method named GridMask based on structured information removal. The proposed method generates a set of binary masks with a sequence of square blocks, with pixel value equal to 0, uniformly distributed over the masks in a grid structure. The squares have the same size and same distance between them. When a mask is applied over the image, the image's regions corresponding to a square region in the GridMask are erased, generating a black region in the image without any information. Chen et al. [18] demonstrated that the proposed method could be successfully applied to different deep learning problems like image classification, object detection, and semantic segmentation. Also, they showed that GridMask outperforms popular erase-based data augmentation techniques like random erasing [19], CutOut [20], and hide-and-seek (HaS) [21].

Recent advances in backbone Convolutional Neural Networks (CNNs) has improved multi-scale representation, leading to consistent performance gains on a wide range of applications. However, most existing methods represent the multi-scale features in a layer-wise manner. Recently, Gao et al. [12] proposed a novel building block for CNNs, a Residual Network (ResNet), namely Res2Net, that can represent multi-scale features at a granular level and increases the range of receptive fields for every network layer. This was achieved by constructing hierarchical residual-like connections within one single residual block. In their work [12], Gao et al. demonstrated that this backbone provides consistent performance gains in multiple computer vision tasks, such as object detection, semantic segmentation, and salient object detection.

To evaluate the Res2Net in SOD, Gao et al. [12] replaced the ResNet-50 backbone in the state-of-the-art Neural Network PoolNet [11] and kept all the other configurations unchanged. The authors then trained a baseline model with the ResNet-50 and a Res2Net model using the MSRA-B dataset [22]. The Res2Net model achieved better results when compared with the ResNet-50 in four famous SOD datasets: ECSSD, PASCAL-S, HKU-IS, and DUT-OMRON.

Proposed by Liu et al. [11], the PoolNet is based on a Feature Pyramid Network (FPN) [23]. Liu et al. [11] also showed that their PoolNet architecture achieves better results with the ResNet-50 backbone than with the VGG-16. Thus, discouraging future works using the PoolNet architecture with a VGG-16 backbone. Following [12], [17], we focus on the PoolNet model without joint training with edge detection.

III. Proposed Work

Our proposed method comprises five steps: background generation, feature extraction, distance optimization, affine transformation, and linear combination. Fig. 1 presents the complete pipeline of our method.

In the first step, we use an inpainting method to remove the salient object from an image altogether, maintaining only the background of the image (details presented in Section III-A). We then extract color and texture features from each new background image and each of the dataset's initial salient objects. In the third step, we compute the distance between the feature vectors to perform an inter-image optimization using the K-Nearest Neighbors (kNN) algorithm and cosine similarity to match an object and a background (details presented in Section III-B). Later on, an intra-image optimization is performed to match a patch of the background and the object.

In the last step, we apply affine transformations in the salient object. The transformations are based on the size and position distribution of the original dataset objects. Then, a linear combination is performed to blend the transformed salient object in the new background, thus generating a new image (details presented in Section III-C).

To showcase what the resulting synthetic images look like, we present in Figure 2 six different examples of the images produced by the IDA method, with input images from the DUTS-TRAINING set. The selection of background images was based on the inter-image distance computation step described in Section III-B.

A. Background Image Generation

The first step of the method is to generate background images, without any salient object, for each image in the training dataset. The idea is that each labeled image can produce both a foreground and a background sample. The foreground image retains only the object of interest, and the background image retains all other information except the data of the object. This introduced void is filled with a technique called image inpainting, producing a complete image.

Image inpainting is the process of restoring missing pixels of digital images plausibly, reconstructing a region based on the surrounding information. Recently, research in image inpainting has received considerable attention in different areas, and it can be used in many different applications, such as restoration of old and damaged documents, removal of unwanted objects, and retouching [24]. Previous works [17] used an UNet-like architecture [25] named PConv [26] to construct background images without salient regions within. We use a neural network called DeepFill v2 [27], [28], officially

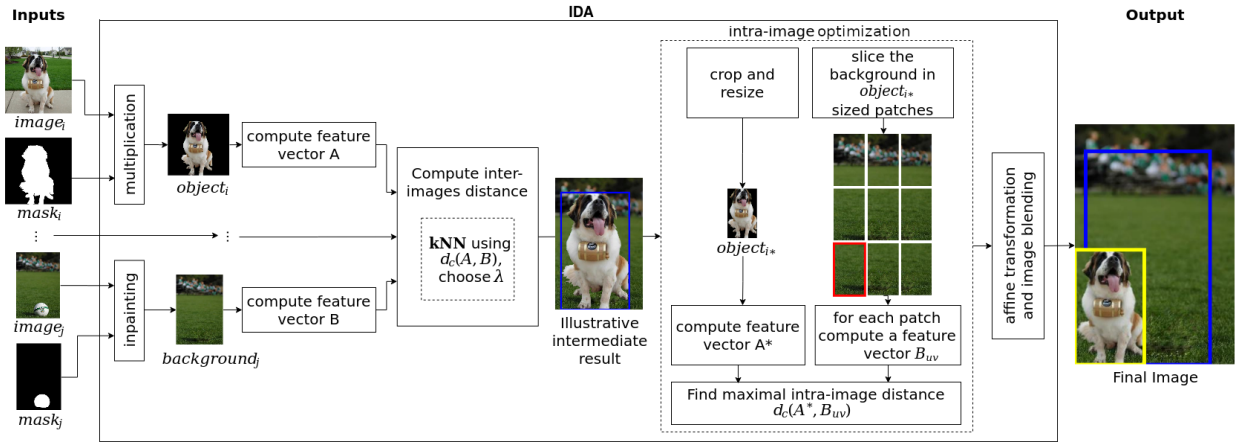
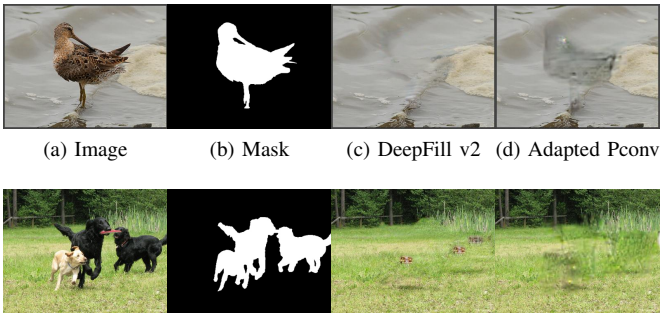


Fig. 1. Flowchart of the IDA technique. The first step is the computation of a background image for all the images of the inputted dataset; an object image is also computed; those images are used to produce feature vectors that can be compared using kNN with cosine similarity; λ stands for the criteria chosen, further details on Section III-B. Highlighted in **blue** is the intermediate bounding box of the object before the affine transformation. An intra-image optimization determines the translation of the object in the new background. Highlighted in **red** is the patch that produces the maximal distance intra-image in this example. Finally, for each pair of $object_i$ and $background_j$, a final image is created using a linear combination of both. Highlighted in **yellow** is the final bounding box of the object.

Fig. 2. Six examples of generated images by IDA. In each bracket, the new image (on the right) was generated with the object of the top-left image and the background of the bottom-left image.



(a) Image (b) Mask (c) DeepFill v2 (d) Adapted Pconv

(e) Image (f) Mask (g) DeepFill v2 (h) Adapted Pconv

Fig. 3. Qualitative difference between inpainted performed by the previous method Adapted Pconv [17] (Subfigures 3d,3h) and the new one DeepFill v2 [27], [28] (Subfigures 3c,3g). Note how the DeepFill v2 method produces a more detailed erasure while sometimes producing an artifact (Subfigure 3g) on the image.

available on[2] and pre-trained with the Places2 dataset [5]. The DeepFill v2 achieved better inpainting results than the adapted version of PConv, as presented in Fig. 3.

B. Feature Extraction and Inter-image Optimization

After generating images without any salient object, our method performs the selection of a new salient object to be inserted in the generated background image. However, when a salient object is inserted in a new random background, the salient object may no longer be salient. Our method chooses a background that retains some of the object's salience to overcome such a problem. To find such a background, we compute a feature vector of 256 positions composed of four histograms for both the salient object and the new background. To represent the color, we employ an HSV color space histogram divided into 64 bins for Hue, 64 for Saturation, 64 for Value, or Brightness. To represent texture, we employ a Local Binary Patterns (LBP) [29] histogram with 64 bins, with the following parameters: the number of circularly symmetric neighbor points set to 24, the circle radius set to 3, and the uniform method.

Then, the kNN algorithm is applied to measure each salient object's distance to each inpainted background. We use $k = 10,553$, in which $10,553$ is the number of images in the DUTS-TRAINING dataset. In the kNN, we use the cosine similarity defined by Equation 1. The similarity value is obtained using the feature vector A originated from the salient object without its original background, and the feature vector B originated from the new background image. In this way, $d_c(A, B)$ shows the similarity between the object and the background.

$$d_c(A, B) = \frac{\sum_{j=1}^{256} A_j B_j}{\sqrt{\sum_{j=1}^{256} A_j^2}\sqrt{\sum_{j=1}^{256} B_j^2}} \quad (1)$$

Given an object o, let μ_o be the mean similarity value of

[2]https://github.com/JiahuiYu/generative_inpainting

the kNNs of o, and σ_o the similarity standard deviation of the kNNs of o. Instead of using the $\lfloor \frac{k}{2} \rfloor$th neighbor for all the images as in [17], a dynamic choice for each image based on the closest neighbor to $\mu_o + \sigma_o$ can better preserve the salience on the new samples. Fig. 4 illustrates the difference when using the $\lfloor \frac{k}{2} \rfloor$ and the $c = closest(\mu_o + \sigma_o)$ criteria.

(a) Mask (b) $\lambda = b_{\lfloor \frac{k}{2} \rfloor}$ (c) $\lambda = b_c$

Fig. 4. Qualitative impact of the background choosing. λ stands for the criteria chosen, b stands for the neighbor chosen. Note how the introduced object is more salient in Subfigure 4c than in Subfigure 4b.

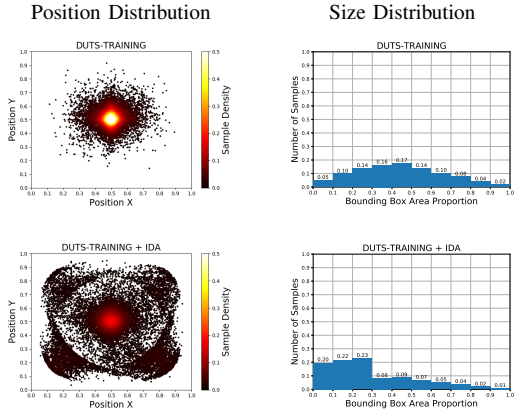


Fig. 5. Position and size distribution per dataset for the training: DUTS-TRAINING, DUTS-TRAINING+IDA. The position distribution is demonstrated in a scatter plot of the normalized bounding box center coordinates. Additionally, a heat colormap represents the sample position density. For size, the bounding box area divided by image area is displayed in a 10-bin histogram.

C. Size distribution and intra-image position optimization

When analyzing the salient datasets available in the literature, we noticed a large difference regarding the salient objects' size and position on the images of training and testing datasets. Fig. 5 and Fig. 6 show the position (left) and size (right) distribution of several dataset utilized in our work.

The position distribution shows each object bounding box's central location in a given dataset, while the size distribution shows the proportion of each salient object to the image size. In Fig. 5, the first row presents the position and size of the DUTS [10] training set, utilized in our work as a baseline training dataset, and in the second row, the DUTS training set combined with our proposal augmentation, respectively. Fig. 6 presents the position and size distribution of testing datasets DUT-OMRON [30], THUR15K [31], HKU-IS [32], DUTS-TEST [10], PASCAL-S [33], and ECSSD [34], respectively.

As presented on Fig. 5, the salient objects in the DUTS dataset, utilized for training, are often located in the center of the image and take between 20 and 60 percent of the image, while the salient objects in other datasets are more widespread in the image. Also, datasets like DUT-OMRON, THUR, PASCAL-S, and DUTS-TEST (Fig. 6), have a higher occurrence of small objects. We generated more examples of small salient objects with different positions in the image with our method, which made the training dataset less biased.

The training dataset is composed of k images. The index i represents the ith image inside the training dataset. Each image i has an object o, a background b, and a bounding box ϕ defined by two corner points $p = (x_{\min}, y_{\min}) = (x, y)$ and $p' = (x_{\max}, y_{\max}) = (x', y')$. The bounding box has a width $w_\phi = x' - x$, a height $h_\phi = y' - y$, and an area $a_\phi = w_\phi \times h_\phi$. The background has a width w_b, a height h_b, and an area $a_b = w_b \times h_b$.

Aiming to resize the object o, we multiply its width and height by a scale factor s_i as in Equation 2. If the resized object o' with $w'_o = w_o \times s_i$ and $h'_o = h_o \times s_i$ can fit b, then we proceed to the next step.

$$s_i = \sqrt{\frac{f(i \bmod 3) \times a_b}{a_\phi}} \tag{2}$$

Let R be a random variable following a uniform distribution with its range is defined by the function $f(i \bmod 3)$ as in Equation 3. Those ranges were chosen to encourage a reduction in size since there are fewer samples with small objects on the DUTS training dataset.

$$f(i \bmod 3) = \begin{cases} R \in [0.075, 0.1), i \bmod 3 = 0; \\ R \in [0.1, 0.2), i \bmod 3 = 1; \\ R \in [0.2, 0.3), i \bmod 3 = 2. \end{cases} \tag{3}$$

If the resized object o' cannot fit b, then there are two cases. In the first case, both ϕ and b have the same orientation, landscape, or portrait, then s is defined as in Equation 4.

$$s_i = 0.5 \times \min\left(\frac{w_b}{w_\phi}, \frac{h_b}{h_\phi}\right) \tag{4}$$

The scalar s_i receives half of the highest possible value to resize the object without deforming it or causing the object to overflow the background. The second case, ϕ, and b have a different orientation, then o is rotated by $-90°$ degrees or $90°$ degrees, the sign of the value is chosen randomly, then the same resize as the first case occurs.

Ruiz *et al.* [17] proposed a random uniform translation on the resized object o' to diversify the position distribution. While this approach can lead to widespread in position distribution it have no way of predicting if the new location preserve salience. Instead of a random uniform translation we propose an intra-image optimization as follows: given the resized object o' and the background b compute the feature vector A^* of o' as described in Section III-B. Slice b in o' sized patches b_{uv}, u and v are the patches indexes. Regarding the margins, ensure that the patch still has $h'_o \times w'_o$. Then, compute

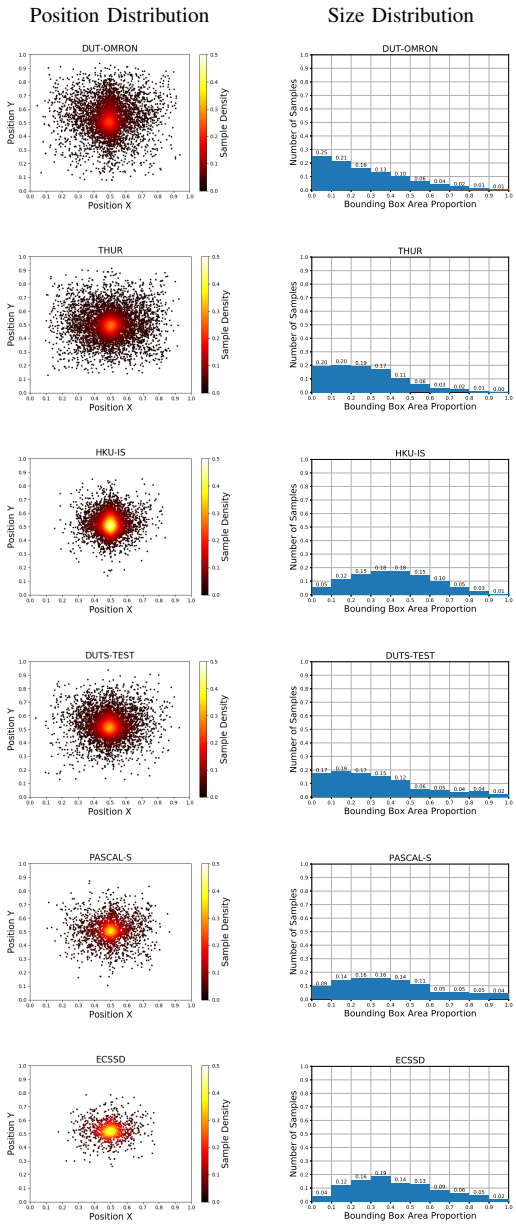


Fig. 6. Position and size distribution per dataset for testing: DUT-OMRON, THUR, HKU-IS, DUTS-TEST, PASCAL-S, and ECSSD. The position distribution is demonstrated in a scatter plot of the normalized bounding box center coordinates. Additionally, a heat colormap represents the sample position density. For size, the bounding box area divided by image area is displayed in a 10-bin histogram.

for every patch b_{uv} a feature vector B_{uv} also the same way as described in Section III-B. Finally, find the maximal distance $d_c(A^*, B_{uv})$, the uv coordinate that maximizes the distance is the patch that o' will superimpose. Fig. 1 illustrates the intra-image optimization.

IV. Experiments

The majority of our experiments were performed in a single state-of-the-art neural network, the PoolNet Res2Net-50. For a fair comparison, we also present the results with the standard PoolNet ResNet-50.

A. Datasets

In our work, we used seven SOD datasets widely used in the literature: DUTS [10], DUT-OMRON [30], THUR15K [31], HKU-IS [32], ECSSD [34], PASCAL-S [33], and SOC [35]. DUTS is one of the largest datasets available on SOD literature, with 10,553 training images and 5,017 testing images. DUT-OMRON is a large dataset, being composed of 5,168 images with a great variety of objects in challenging backgrounds. THUR15K has 6,232 labeled images divided into five categories: butterfly, coffee mug, dog jump, giraffe, and plane. HKU-IS is composed of 4,447 images with low contrast and multi salient objects. ECSSD has 1,000 images with a high content variety and different objects. PASCAL-S derives from the validation set of the PASCAL VOC 2012 [36] dataset and is composed of 2,205 images. Salient Objects in Clutter (SOC) is a challenging dataset that, differently from other SOD datasets, includes images with no salient objects and includes salient objects with real-world challenging visual occurrences like motion blur, occlusion, and cluttered background.

B. Metrics

To evaluate our data augmentation technique's impact, we use five evaluation metrics: Precision, Recall, F-score, Mean Absolute Error (MAE), and the Structure-measure [37].

In the F-score, we use β equals to 0.3 as done in [11], [17], in order to weight precision more than recall. It is important to state that these metrics are designed to compare two binary maps and the salience maps evaluated are non-binary. For that reason, binarization is necessary for us to use those metrics. The binarization process is threshold-dependent, so we follow the evaluation procedure used on [11] that defines F_β^* (Equation 7), P^* (Equation 8), R^* (Equation 9).

$$P_\mu(th) = \frac{1}{N} \times \sum_{i=1}^{N} P(th, i) \tag{5}$$

$$R_\mu(th) = \frac{1}{N} \times \sum_{i=1}^{N} R(th, i) \tag{6}$$

$$F_\beta^* = max(\{\frac{(1+\beta^2) \times P_\mu(th) \times R_\mu(th)}{\beta^2 \times P_\mu(th) + R_\mu(th)} | 0 < th < 255\}) \tag{7}$$

N is the number of images, $P(th, i)$ and $R(th, i)$ are the Precision of an image binarized i with threshold th, and the Recall with the same parameters, respectively. Finally, $bestTh$ is the threshold th that produces the maximal F_β^*.

$$P^* = P_\mu(bestTh) \tag{8}$$

$$R^* = R_\mu(bestTh) \tag{9}$$

Structure-measure, or simply s-measure, proposed on [37], evaluate non-binary foreground maps. This metric simultaneously evaluates region-aware and object-aware structural similarity between a Salience map and a Ground Truth map.

C. Quantitative Results

We present our findings in this subsection after training the PoolNet Res2Net50 neural network with different data augmentation techniques. Table I shows the comparison between our baseline architecture, PoolNet with ResNet-50 as a backbone, trained with one data augmentation technique, the PoolNet with Res2Net-50 as the backbone, trained with no data augmentation; and other six training sets, each with a different data augmentation technique.

F-measure, Precision, Recall were not computed for the SOC dataset, as was recommended by Fan et al. [35], since SOC contains many non-salient images, for such, the ground-truth is an all-zero matrix, thus directly using the F-measure may result in an inaccurate score.

Flip operations are almost always used as data augmentation due it is simplicity and effectiveness [11], [14]–[16]. So, here H-Flip was chosen to represent how our method compares with one that uses affine transformations only and to ensure a fair comparison with results presented by [11]. GridMask was chosen to represent how a data erasure method compares with ours. To measure the impact of the newly proposed improvements, we also present a comparison with our previous work ANDA [17].

TABLE I
COMPARISON BETWEEN BASELINE AND DATA AUGMENTED RESULTS. THE BEST F_β^* IS HIGHLIGHTED IN **BOLD BLUE TEXT**. THE BEST S-SCORE IS HIGHLIGHTED IN **BOLD TEXT**. DUT-O* IS AN ABBREVIATION OF DUT-OMRON. DUTS-TE* IS AN ABBREVIATION OF DUTS-TEST.

Experiment	Metric	DUT-O*	THUR15K	PASCAL-S	DUTS-TE*	HKU-IS	ECSSD	SOC
	N# Images	**5,168**	**6,232**	**2,205**	**5,017**	**4,447**	**1,000**	**6,000**
PoolNet ResNet-50 Baseline H-Flip Only [11]	S-score↑	0.8341	0.8347	0.8368	0.8824	0.9152	0.9207	0.8464
	F_β^*↑	0.8305	0.8026	0.8716	0.8858	0.9343	0.9444	-
	P^*↑	0.8602	0.7945	0.9058	0.9104	0.9583	0.9649	-
	R^*↑	0.7448	0.8309	0.7744	0.8128	0.8624	0.8821	-
	MAE↓	0.0554	0.0698	0.0733	0.0397	0.0327	0.0388	0.0913
No data aug	S-score↑	0.8219	0.8369	0.8313	0.8760	0.9087	0.9202	0.8623
	F_β^*↑	0.8391	0.8092	0.8714	0.8880	0.9318	0.9450	-
	P^*↑	0.8872	0.8023	0.9086	0.9197	0.9585	0.9670	-
	R^*↑	0.7106	0.8331	0.7668	0.7964	0.8526	0.8785	-
	MAE↓	0.0513	0.0650	0.0726	0.0376	0.0321	0.0365	0.0727
GridMask Only	S-score↑	0.8271	0.8341	0.8359	0.8774	0.9112	0.9232	0.8476
	F_β^*↑	0.8333	0.8040	0.8700	0.8848	0.9327	0.9471	-
	P^*↑	0.8709	0.7957	0.9034	0.9141	0.9574	0.9650	-
	R^*↑	0.7285	0.8332	0.7744	0.7993	0.8588	0.8917	-
	MAE↓	0.0526	0.0684	0.0717	0.0388	0.0321	0.0355	0.0881
ANDA Only [17]	S-score↑	0.8337	0.8355	0.8372	0.8801	0.9161	0.9242	0.8515
	F_β^*↑	0.8289	0.8048	0.8686	0.8845	0.9346	0.9463	-
	P^*↑	0.8544	0.7912	0.9013	0.9103	0.9571	0.9650	-
	R^*↑	0.7539	0.8539	0.7750	0.8081	0.8668	0.8890	-
	MAE↓	0.0572	0.0692	0.0727	0.0404	0.0305	0.0352	0.0851
IDA Only	S-score↑	0.8362	0.8393	0.8351	0.8831	0.9144	0.9244	0.8623
	F_β^*↑	0.8371	0.8102	0.8714	0.8870	0.9336	**0.9475**	-
	P^*↑	0.8723	0.8018	0.9068	0.9108	0.9565	0.9666	-
	R^*↑	0.7376	0.8397	0.7710	0.8159	0.8645	0.8891	-
	MAE↓	0.0538	0.0667	0.0722	0.0387	0.0314	0.0363	0.0748
H-Flip Only	S-score↑	0.8350	0.8439	0.8376	0.8850	0.9152	0.9240	0.8651
	F_β^*↑	0.8392	0.8133	0.8735	0.8938	0.9369	0.9471	-
	P^*↑	0.8791	0.8025	0.9067	0.9194	0.9625	0.9670	-
	R^*↑	0.7292	0.8517	0.7784	0.8178	0.8607	0.8863	-
	MAE↓	0.0518	0.0649	0.0719	0.0371	0.0312	0.0365	0.0729
GridMask + H-Flip	S-score↑	0.8369	0.8415	**0.8391**	0.8862	0.9156	0.9229	0.8472
	F_β^*↑	0.8398	0.8135	0.8730	0.8930	0.9345	0.9459	-
	P^*↑	0.8843	0.8086	0.9150	0.9226	0.9626	0.9651	-
	R^*↑	0.7193	0.8300	0.7571	0.8066	0.8517	0.8869	-
	MAE↓	0.0555	0.0682	0.0743	0.0396	0.0340	0.0404	0.0932
IDA + GridMask + H-Flip	S-score↑	**0.8441**	**0.8449**	0.8382	**0.8918**	**0.9190**	**0.9256**	**0.8711**
	F_β^*↑	**0.8444**	**0.8167**	**0.8762**	**0.8989**	**0.9386**	0.9468	-
	P^*↑	0.8910	0.8130	0.9145	0.9268	0.9641	0.9640	-
	R^*↑	0.7191	0.8294	0.7690	0.8170	0.8626	0.8936	-
	MAE↓	0.0540	0.0663	0.0737	0.0377	0.0324	0.0388	0.0710

We concluded that the drastic change of backbone architecture only improves the overall results by a small margin, i.e., for F-measure a gain of 1.07 on the best case and 0.19 on the worst-case comparing the ResNet-50 (baseline with H-flip) with the Res2Net-50 H-flip only.

After the backbone comparison, we analyzed how much the data augmentation can further improve those gains. Our method alone (IDA Only on Table I) was not enough to surpass the traditional H-Flip but did surpass the ANDA only and GridMask Only in F-measure in five of six datasets. We also experimented with a combination of techniques.

Maintaining the same architecture (Res2Net-50), we further improved the results by 0.52 on the best case, -0.03 in the worst case, comparing Res2Net50 H-flip only to IDA (Ours) + GridMask + H-Flip, this improvement was made by only applying data augmentation techniques surpassing the previous results on these datasets [11], [12].

The combination of our proposed method IDA with GridMask and H-Flip achieved the best s-measure in all evaluated datasets (except in the PASCAL-S), and best F-score in all datasets except the ECSSD in which the IDA only achieved the best F-score. In the case of precision, it achieved the best result in four of the six datasets.

D. Ranking models

To rank the performance of the saliency models we use an average ranking technique. Similarly to [38], this approach can summarize multiple metrics into a single, more readable value. The set of test images were generated by combining all testing datasets except SOC, since it contains images without any salience, resulting in a total of $24,069$ images. Nine metrics were chosen to compute the ranking of each model:

1) S-measure
2) F_β
3) Precision (Pr)
4) Recall (Re)
5) MAE
6) Specificity (Sp)
7) False Positive Rate (FPR)
8) False Negative Rate (FNR)
9) Percentage of Wrong Classifications (PWC)

The average ranking of a saliency model M_i is described by Equation 10. τ is a set of images in a test dataset, and P is the number of metrics utilized to evaluate the model M. To ensure a fair comparison using binary metrics, we use the same fixed threshold for every model.

$$R_i = \frac{1}{P}\sum_{j=1}^{P} rank(metric_j(M_i(\tau)); metric_j(M_k(\tau)), \forall k \neq i) \tag{10}$$

For metrics which higher values are desirable a descending sort is employed, e. g., using a $metric_j$ the models M_0, M_1, M_2 achieved the values $[0.97, 0.94, 0.95]$ respectively, so M_0 is ranked as 1, M_2 as 2, M_1 as 3. An ascending order is employed otherwise. In both cases, models with rank close to one are considered better.

As presented in Table I, the IDA, when combined with other techniques, achieved the best results in the majority of test

datasets. So, in this section, we present further variations of training and the results, which are presented in Table II. Three variations of the (IDA + GridMask + H-flip) approach were evaluated, and are described as follows:

- In (IDA+Grid+H-flip)* we use a proportion of 2 original images to 1 IDA image, resulting in training 31,659 images; this makes the synthetic images have a smaller impact on the training.
- (IDA+Grid+H-flip)′ stands for a reduction of samples generated from IDA; the 611 top worst f-measure training samples were removed from the training set, resulting in 20,495 images. The evaluation to find the worst samples was done using a pre-trained model.
- (IDA+Grid+H-flip)″ stands for the same as (IDA+Grid+H-flip) with the difference that, instead of using a fixed probability of applying the GridMask in all steps, the probability increases accordingly to the epoch during training, it starts with a probability of 0.0 at the first epoch and ends with a probability of 1.0 at the final epoch.

TABLE II
AVERAGE RANKING BETWEEN THE DATA AUGMENTATIONS TECHNIQUES. OUR METHOD, COMBINED WITH OTHERS (IDA+GRID+H-FLIP) ACHIEVED THE RANK CLOSER TO ONE, GETTING AHEAD OF THE OTHER METHODS. THE TEST IMAGES WERE GENERATED BY COMBINING ALL TESTING DATASETS EXCEPT SOC, DUE TO THE IMAGES WITHOUT ANY SALIENCE, WITH A TOTAL OF 24,069 IMAGES.

Data Augmentation Technique	Average ranking
IDA+Grid+H-flip	2.7
(IDA+Grid+H-flip)′	4.1
IDA+H-flip	4.4
H-flip only	5.1
(IDA+Grid+H-flip)*	6.2
Grid+H-flip	6.3
(IDA+Grid+H-flip)″	6.3
IDA only	6.5
No data Augmentation	7.7
ANDA Only	8.1
Grid Only	8.2

The ranking procedure considers multiple metrics that can describe different aspects of the resulting segmentation masks and summarize them, making for a more readable result. Three different models containing our generated images were ranked above the standard horizontal flip and the GridMask method, meaning that the results were superior in multiple metrics.

E. Qualitative results

An example of each testing dataset and the qualitative difference between segmentation performed with different training data augmentation techniques is presented in Fig. 7. Three different approaches are illustrated: the best overall model for the data augmentation that uses the synthetic sample of IDA, our method, Random Horizontal Flip, and the GridMask method (Fig. 7 third column); the standard data augmentation technique, horizontal flip (Fig. 7 fourth column); and the one produced by the previous technique ANDA (Fig. 7 fifth column).

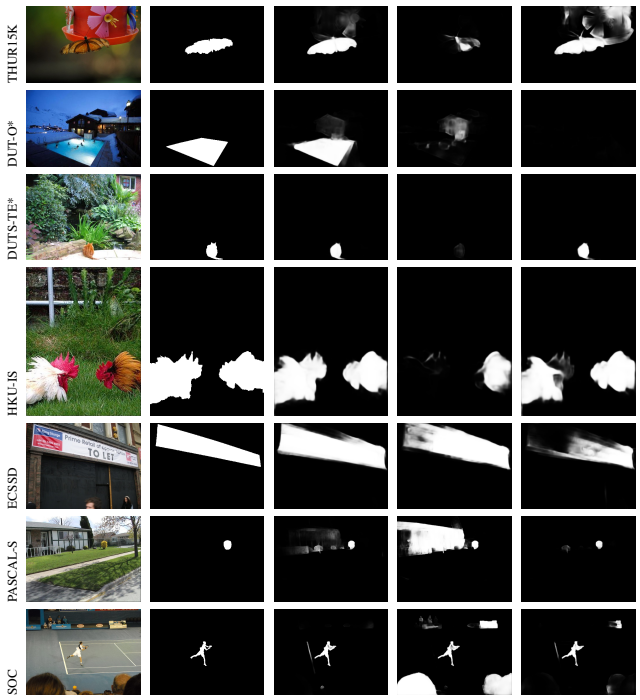


Fig. 7. Qualitative difference between segmentation performed with different training data augmentation techniques. An example of each testing dataset is presented. IDA* refers to the best overall model and stands for the data augmentation model that uses the synthetic sample of IDA, our method, Random Horizontal Flip, and the GridMask method. DUT-O* is an abbreviation of DUT-OMRON. DUTS-TE* is an abbreviation of DUTS-TEST. GT is an abbreviation of Ground Truth.

Note how using the model with IDA, the resulting segmentation improves both in reducing false positives as in the SOC example (Fig. 7 last row) and increasing true positives like in the HKU-IS (Fig. 7 fourth row). However, it still produced some false negatives with a low confidence level, represented by a low grayscale value. This is exemplified at the subfigure of THUR15K (Fig. 7 first row).

V. CONCLUSION

In this paper, we propose a new data augmentation technique named IDA in the context of SOD. It uses a linear combination of two different images, the resulting image contains in the foreground a salient object segmented from its original background, affinely transformed, and a full background created using image inpainting to erase its labeled objects. The background choice is based on an inter-image optimization, while object size follows a uniform random distribution within a specified interval, and the object position is intra-image optimal.

During our experiments, two offline (in which images were processed before training started), and two online (in which images were processed per batch during training), data augmentation techniques were analyzed. The offline ones were the ANDA and IDA, while the online ones were the GridMask and

random horizontal-flip. The GridMask method has shown an unstable behavior, sometimes improving the results and some times worsening it. Such behavior could be directly tied to its parameterization, in which changing the ratio of the squares delete more or less information. Removing a large chunk of information in a small object can damage the learning process. The methods that achieved the highest results included this policy, meaning that structured information removal can encourage generalization while somewhat unstable. On the other hand, the horizontal flip operation consistently provides an improvement in the results and has a single parameter, the probability of application.

Training a neural network model with synthetic images generated by IDA and other online data augmentation techniques provided a consistent improvement of the S-measure and F-score results in six out of seven test datasets. The achieved results surpassed the previously reported results, setting a new state-of-the-art. These results show that our method is a relevant contribution to the data augmentation training techniques. Furthermore, it can be combined with other online data augmentation techniques providing a more robust policy that can improve deep neural networks training due to their introduced information variance, which encourages the model to learn more generic features and reduce overfitting.

ACKNOWLEDGMENT

The authors thank the Coordination for the Improvement of Higher Education Personnel (CAPES) for the Masters scholarship. We gratefully acknowledge the founders of the publicly available datasets and the support of NVIDIA Corporation with the donation of the GPUs used for this research.

REFERENCES

[1] C. Koch and S. Ullman, *Shifts in Selective Visual Attention: Towards the Underlying Neural Circuitry*. Dordrecht: Springer Netherlands, 1987, pp. 115–141.

[2] W. Wang, Q. Lai, H. Fu, J. Shen, and H. Ling, "Salient object detection in the deep learning era: An in-depth survey," *arXiv:1904.09146*, 2019.

[3] B. A. Krinski, D. V. Ruiz, G. Z. Machado, and E. Todt, "Masking salient object detection, a mask region-based convolutional neural network analysis for segmentation of salient objects," in *Latin American Robotics Symposium (LARS)*, 2019, pp. 55–60.

[4] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR*, 2009.

[5] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE TPAMI*, vol. 40, no. 6, pp. 1452–1464, June 2018.

[6] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, ser. NIPS'14, 2014, pp. 3320–3328.

[7] C. Shorten and T. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, 12 2019.

[8] D. V. Ruiz, G. Salomon, and E. Todt, "Can giraffes become birds? an evaluation of image-to-image translation for data generation," *preprint arXiv 2001.03637*, 2020.

[9] A. Borji, M.-M. Cheng, Q. Hou, H. Jiang, and J. Li, "Salient object detection: A survey," *Computational Visual Media*, vol. 5, no. 2, p. 117–150, Jun 2019.

[10] L. Wang, H. Lu, Y. Wang, M. Feng, D. Wang, B. Yin, and X. Ruan, "Learning to detect salient objects with image-level supervision," in *CVPR*, 2017.

[11] J.-J. Liu, Q. Hou, M.-M. Cheng, J. Feng, and J. Jiang, "A simple pooling-based design for real-time salient object detection," in *CVPR*, 2019.

[12] S.-H. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang, and P. Torr, "Res2net: A new multi-scale backbone architecture," *IEEE TPAMI*, 2020.

[13] F. Perazzi *et al.*, "Learning video object segmentation from static images," in *CVPR*. IEEE, 2017.

[14] L. Guo and S. Qin, "High precision detection of salient objects based on deep convolutional networks with proper combinations of shallow and deep connections," *Symmetry*, vol. 11, no. 1, 2018.

[15] J. Wei, S. Wang, and Q. Huang, "F3net: Fusion, feedback and focus for salient object detection," *arXiv:1911.11445*, 2019.

[16] B. Wang, Q. Chen, M. Zhou, Z. Zhang, X. Jin, and K. Gai, "Progressive feature polishing network for salient object detection," *arXiv:1911.05942*, 2019.

[17] D. V. Ruiz, B. A. Krinski, and E. Todt, "Anda: A novel data augmentation technique applied to salient object detection," in *International Conference on Advanced Robotics (ICAR)*, 2019, pp. 487–492.

[18] P. Chen, S. Liu, H. Zhao, and J. Jia, "Gridmask data augmentation," *arXiv:2001.04086*, 2020.

[19] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," in *AAAI*, 2020.

[20] T. DeVries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout," *arXiv:1708.04552*, 2017.

[21] K. K. Singh, H. Yu, A. Sarmasi, G. Pradeep, and Y. J. Lee, "Hide-and-seek: A data augmentation technique for weakly-supervised localization and beyond," in *arXiv:1811.02545*, 2018.

[22] T. Liu, J. Sun, N. Zheng, X. Tang, and H. Shum, "Learning to detect a salient object," in *CVPR*, 2007, pp. 1–8.

[23] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *CVPR*, 2017, pp. 936–944.

[24] M. A. Qureshi, M. Deriche, A. Beghdadi, and A. Amin, "A critical survey of state-of-the-art image inpainting quality assessment metrics," *Journal of Visual Communication and Image Representation*, vol. 49, pp. 177 – 191, 2017.

[25] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention*, 2015, pp. 234–241.

[26] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao, and B. Catanzaro, "Image inpainting for irregular holes using partial convolutions," in *Computer Vision (ECCV)*, 2018, pp. 89–105.

[27] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *CVPR*, 2018.

[28] ——, "Free-form image inpainting with gated convolution," in *ICCV*, 2019.

[29] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, July 2002.

[30] C. Yang, L. Zhang, H. Lu, X. Ruan, and M. Yang, "Saliency detection via graph-based manifold ranking," in *CVPR*, 2013, pp. 3166–3173.

[31] M.-M. Cheng, N. J. Mitra, X. Huang, and S.-M. Hu, "Salientshape: Group saliency in image collections," *Vis. Comput.*, vol. 30, no. 4, p. 443–453, Apr. 2014.

[32] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *CVPR*, 2015.

[33] Y. Li, X. Hou, C. Koch, J. M. Rehg, and A. L. Yuille, "The secrets of salient object segmentation," in *CVPR*, 2014.

[34] J. Shi, Q. Yan, L. Xu, and J. Jia, "Hierarchical image saliency detection on extended cssd," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 4, pp. 717–729, April 2016.

[35] D.-P. Fan, M.-M. Cheng, J.-J. Liu, S.-H. Gao, Q. Hou, and A. Borji, "Salient objects in clutter: Bringing salient object detection to the foreground," in *ECCV*. Springer, 2018.

[36] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results," http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

[37] D.-P. Fan, M.-M. Cheng, Y. Liu, T. Li, and A. Borji, "Structure-measure: A new way to evaluate foreground maps," in *ICCV*, 2017.

[38] M. A. Contreras-Cruz, D. E. Martinez-Rodriguez, U. H. Hernandez-Belmonte, and V. Ayala-Ramirez, "A genetic programming framework in the automatic design of combination models for salient object detection," *Genetic Programming and Evolvable Machines*, vol. 20, no. 3, pp. 285–325, 2019.