SLICE: Stabilized LIME for Consistent Explanations for Image Classification

Revoti Prasad Bora [†] Philipp Terhörst[‡] Raymond Veldhuis[†]
Raghavendra Ramachandra[†] Kiran Raja[†]
[†] Norwegian University of Science and Technology, Norway
[‡]University of Paderborn, Germany
{revoti.p.bora; raymond.veldhuis, raghavendra.ramachandra, kiran.raja} @ntnu.no
philipp.terhoerst@uni-paderborn.de

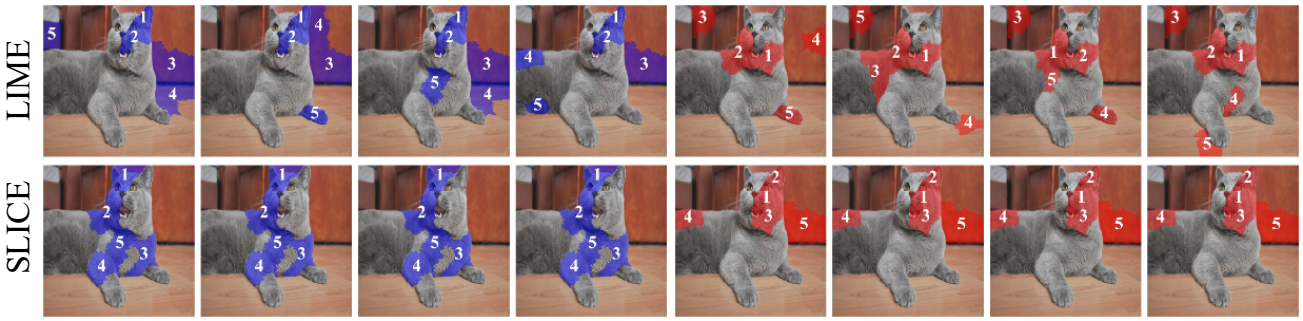


Figure 1. Representative explanations of Inception v3 model's prediction by LIME and SLICE(proposed) for a random image from Oxford-IIIT pets dataset. The prediction class was Egyptian cat with a probability of 0.45. (Top 5 positive superpixels are marked in blue segments and top 5 negative are marked in red). The ranks are indicated on the superpixels (lower number signifies higher importance). For LIME, some of the superpixels identified as positive in one run are identified as negative in another run. In contrast, all superpixels marked as positive or negative never change signs for SLICE. The relative importance ranks of superpixels in SLICE are stable across all the runs.

Abstract

Local Interpretable Model-agnostic Explanations (LIME) - a widely used post-ad-hoc model agnostic explainable AI (XAI) technique. It works by training a simple transparent (surrogate) model using random samples drawn around the neighborhood of the instance (image) to be explained (IE). Explanations are then extracted for a black-box model and a given IE, using the surrogate model. However, the explanations of LIME suffer from inconsistency across different runs for the same model and the same IE. We identify two main types of inconsistencies: variance in the sign and importance ranks of the segments (superpixels). These factors hinder LIME from obtaining consistent explanations. We analyze these inconsistencies and propose a new method, Stabilized LIME for Consistent Explanations (SLICE). The proposed method handles the stabilization problem in two aspects: using a novel feature selection technique to eliminate spurious superpixels and an adaptive perturbation technique to generate perturbed images in the neighborhood of IE. Our results demonstrate that the explanations from SLICE exhibit significantly better consistency and fidelity than LIME (and its variant BayLime).

1. Introduction

In the broad spectrum of post-ad-hoc explanation methods, model-agnostic methods like LIME [13], SHAP [9] and their variants have been popular for extracting explanations from Black-Box models. While explanation methods such as Grad-CAM [16], Grad-CAM++ [2], and Ablation-CAM [12] require access to the intermediate layers of the model, methods like LIME, and SHAP methods require access only to the input and the output of the model.

In our paper, we focus on the category of post-ad-hoc methods that uses local surrogate models for explanations. While LIME has the advantages of being model-agnostic and being able to extract explanations in a post-ad-hoc manner, it is also inconsistent in the explanations [5, 7, 8, 22, 23, 25, 26]. [23] observes three types of uncertainty: sampling variance in explaining a data point, sensitivity to the choice of parameters such as the size of the neighborhood and sample size, and variation of model cred-

ibility across different data points. These uncertainties lead to inconsistency in explanations. [5] and [7] point out the inconsistency of additive explanations and observe the differences in feature importance across various methods.

Fig. 1 depicts the inconsistencies in LIME explanations(upper row). Top 5 superpixels (marked blue), which are indicated to have a positive impact on the output probability in one run, are also indicated to have a negative impact in a different run. We define this uncertainty of the superpixels (flipping between positive and negative directions) as the sign entropy. Similarly, the order of importance indicated by the integers as ranks of the superpixels also varies for both sets of positive and negative superpixels in LIME. We refer to this inconsistency as the relative importance of ranks of superpixels. Both these inconsistencies lead to ambiguity in the explanations. Our work aims at stabilizing LIME to provide consistent explanations under post-ad-hoc category. We propose SLICE to address the inconsistencies mentioned earlier and achieve significant consistency and fidelity in the explanations. (lower row of Fig. 1).

1.1. Our Contributions

We propose SLICE to address the issues of sign entropy and high variance in the importance ranks of superpixels through a comprehensive and integrated approach. Our contributions in this paper are:

1. We present a novel feature selection method to eliminate spurious superpixels, i.e., with high sign entropy to enable consistent explanations (see details in Sec. 3.1).
2. We hypothesize that LIME's inconsistency partly stems from its perturbation method, which creates perturbed images far from the IE. To generate perturbed images closer to the IE, we propose using Gaussian Blur and develop a novel method for adaptively selecting the hyper-parameter σ (see details in Sec. 3.2).
3. We propose a new metric to measure the consistency of explanations that take into consideration the sign entropy and relative importance ranks of superpixels (details in Sec. 3.3).
4. We conduct extensive experimental analysis in evaluating the consistency and fidelity of SLICE explanations and compare it with those of LIME and BayLIME.

2. Existing Works on Stability of LIME

DLIME [22] uses Hierarchical Clustering to partition the dataset into different clusters and select points from cluster nearest to the IE, thus ensuring to adhere to the locality assumption of LIME. S-LIME [26] tries to stabilize LIME explanations by their proposed hypothesis testing framework. This hypothesis testing framework uses the Central Limit Theorem to determine the samples required to guarantee the stability of the explanations. ALIME [17] uses an autoencoder as the weighting function to calculate the distance of samples points from the IE, thereby making the coefficients more stable. BayLIME [25] works by utilizing a Bayesian local surrogate model, which was shown as a Bayesian principled weighted sum of prior knowledge and estimates based on new samples. As our work primarily focuses on image and vision applications, we consider original LIME and BayLIME as state-of-the-art counterparts [1] to compare against our proposed SLICE.

3. Proposed Approach

In this section, we discuss the components of SLICE, i.e., Sign Entropy based Feature Elimination and adaptive selection of sigma hyper-parameter for Gaussian Blur. Further, we also discuss the proposed consistency metric to evaluate the consistency of explanations.

3.1. Sign Entropy based Feature Elimination

We propose a feature (superpixel in our context) selection algorithm as our first major novel contribution. The proposed feature selection algorithm estimates the sign entropy of superpixels and eliminates the features with positive sign entropy. These eliminated superpixels are excluded from the explanations, making the explanations more consistent in terms of sign entropy. The proposed feature selection algorithm eliminates spurious features (superpixels with positive sign entropy). Hence, we will refer to it as Sign Entropy based Feature Elimination (SEFE) subsequently.

SEFE is an iterative approach, and it begins the first iteration by bootstrapping the dataset D obtained by randomly perturbing the IE. In our context, the sampled points are images obtained by randomly perturbing different superpixels. The dataset D is a combination of the random perturbation vectors (to be referred to as X) and the corresponding prediction probability ($\hat{Y}$) for the top prediction class. Although we used the top predicted class in our experiments, any other class can be used without the loss of generality. The coefficients from the trained bootstrapped models are used to estimate the sign entropy of the coefficients using Kernel Density Estimate (KDE). We use Scott's rule of thumb [15] to calculate the bandwidth in all our KDE implementations, as it is well-known and widely used. The coefficients with positive sign entropy are considered unstable, and so are the superpixels associated with these coefficients (explained in Sec. 1).

In the next iteration, the superpixels with positive sign entropy are not perturbed, thereby removing their contribution in predicting ($\hat{Y}$) and the process of estimating sign entropy of coefficients is repeated as mentioned earlier. This process is carried out until a subset of stable superpixels is left or the algorithm has run for a predefined number of iterations. After getting a subset of stable superpixels, the

[1] While there exits variants of LIME such as ALIME, BLIME and BayLIME, only BayLIME is designed to work with images.

algorithm runs for 'max_tolerance' more iterations to ascertain no unstable features in the selected subset. After the algorithm exhausts all iterations, either a subset of stable features is selected or none are identified as stable. If none of the features were identified as stable, then the algorithm can be run with a higher threshold of sign entropy to select relatively stable superpixels. In our experiments, we considered the threshold of sign entropy as 0. In case no superpixels were identified as stable, our algorithm does not eliminate any superpixel and uses Gaussian blur as a perturbation technique with Ridge Regression as the surrogate model (refer Sec. 3.2 and Sec. 5.2). The details of using Gaussian blur and adaptively selecting sigma are discussed in the following subsection. Out of 200 image-model combinations in our experiments, we found only one combination where none of the superpixels were selected as stable (for sign entropy threshold = 0). We show the details of SEFE in Algorithm 1.

Algorithm 1 Sign Entropy based Feature Elimination (SEFE)

Input: Data D, max_tolerance T, number of bootstrap samples $N = 10000$, max iterations $iter\_max = 10$
Output: Dataframe containing coefficients after the removal of eliminated features $\mathbf{C}$.
Initialize: Tolerance counter $t = 0$, final coefficients $\mathbf{C}$
while $t < T$ OR iter $<$ $iter\_max$ **do**
    Initialize a matrix $\mathbf{C}'$ with dimensions $N\times$ number of features in D
    for $i = 1$ to N **do**
        Bootstrap sample D_i from D
        Fit ridge regression model on D_i to get coefficients $\mathbf{c}_i$
        $\mathbf{C}'[i, :] \leftarrow \mathbf{c}_i$
    Compute the sign entropy H for each coefficient across the N models using KDE
    $F' \leftarrow$ coefficients with $H > 0$
    if $|F'| \neq 0$ **then**
        $D \leftarrow D[:, \neg F']$
        $t \leftarrow 0$
    else
        if $t == 0$ **then**
            $\mathbf{C} \leftarrow \mathbf{C}'$
        $t \leftarrow t + 1$
    iter $\leftarrow$ iter $+ 1$
return $\mathbf{C}$

The elimination of features with the possibility of sign flips stabilizes the importance ranks of the superpixels contributing to the consistency of the explanations. Although we used Gaussian blur as the perturbation technique, any other perturbation technique can be used with SEFE. We show the effectiveness of SEFE by assigning 0 to all pixels of a perturbed superpixel (similar to LIME - refer Sec. 5.2).

3.2. Adaptive Gaussian Blur

The perturbation method used in LIME assigns the value of 0 to all pixels of a superpixel. This is a simple way to perturb an image, but it leads to high fluctuations in $(\hat{Y})$ for the perturbed images (see Fig. 2). Further, it can be observed that the variation in $(\hat{Y})$ also increases with an increase in sigma value. Thus, the perturbed images corresponding to the same perturbation vector but with different perturbations can result in a wide range of $(\hat{Y})$. For example, the highest variance in $(\hat{Y})$ is observed for the perturbation of the original LIME implementation and the lowest for sigma = 0.1. Hence, it is crucial to find a perturbation that induces enough variation in $(\hat{Y})$ for the surrogate model to learn but at the same time not too high to violate the locality assumption of LIME.

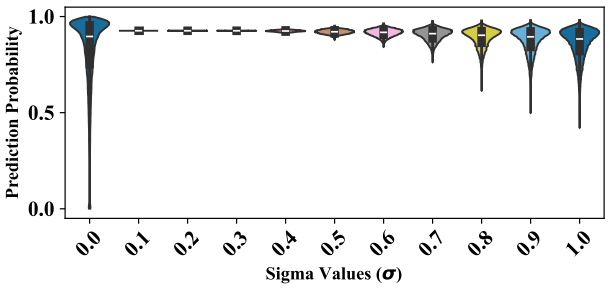


Figure 2. Violin plot of the $\hat{Y}$ corresponding to different sigma (σ) values for a sample image from Oxford-IIIT Pets dataset scored with Inception V3 model (ImageNet weights). A σ of 0 indicates the perturbation used in the original LIME, i.e., assigning 0 to all pixels in the superpixel to be perturbed. All other σ from 0.1 to 1 ($\forall\sigma > 0$) indicate the value of the standard deviation used for perturbing the superpixel with Gaussian Blur. The output probabilities are accumulated across 50 runs with 500 perturbation examples in each run.

Since we are considering local explanations, we examine the interaction of each model with each image separately. We refer to it as an image-model pair. Thus each image-model pair can be considered as a distinct process, where the process is defined as running a particular model on an image perturbed with a specific sigma value. This can be seen in Fig. 2, where different output probabilities are generated for the same perturbation vector for different value of sigma. We then select the sigma value for which the linear regression model (surrogate model) exhibits the maximum value of adjusted R-squared. Adjusted R^2 ($\bar{R}^2$) which is defined as:

$$\bar{R}^2 = 1 - (1 - R^2) \cdot \frac{n-1}{n-p-1},$$

where R^2 is the Coefficient of Determination, n is the number of observations and p is the number of predictors. The term $\frac{n-1}{n-p-1}$ is a penalty term for adding non-significant predictors. $\bar{R}^2$, is an extension of R^2 that penalizes the addition of non-significant predictors. In this context, a high $\bar{R}^2$ value (close to 1) suggests that the model can explain a large portion of the variance in the target variable. Since the surrogate model is a linear model, this suggests a strong linear relationship between the predictors and the target.

This approach ensures that our sigma selection is adaptive, adjusting to the specific needs of each image-model pair, which is crucial, as different images and models may respond differently to Gaussian perturbations, necessitating a flexible and adaptive approach for selecting sigma. We will refer to our method of adaptively selecting the hyper-parameter sigma for Gaussian Blur as Ada-Blur subsequently.

3.3. Proposed Consistency Metric

The proposed consistency metric has two components to address the two aspects of consistency: sign flips of the coefficients and the variance in the rank of the coefficients of the surrogate model. We define both the components below:

1. **Average Sign Flip Entropy (ASFE)**: We propose Average Sign Flip Entropy as a measure of the variability in the sign of a superpixel across multiple runs. It reflects the degree of inconsistency in the direction of the effect of the explanatory variables. A lower value of ASFE would indicate that the concerned superpixel has lower variance in sign (i.e., either positive or negative) across multiple runs. ASFE for model 'Model' and explanation technique 'xp' is calculated as below:

$$ASFE^{xp}_{Model} = \frac{1}{n}\sum_{i=1}^{n} H(\text{sign}_i) \tag{1}$$

Where,

$$H(\text{sign}_i) = -p_i^+ \log_2(p_i^+) - p_i^- \log_2({}_ip^-)$$

where, $H(\text{sign}_i)$ is the sign entropy of the i^{th} superpixel. The quantities p_i^+ and p_i^- are the probabilities of the i^{th} superpixel to be positive or negative respectively. p_i^+ and p_i^- are calculated for each of the 'n' superpixels by using Kernel Density Estimation (KDE) owing to its non-parametric nature[15]. $ASFE^{xp}_{Model}$ ranges between [0,1] where 0 indicates no sign flips and 1 denotes high sign flips of the superpixels.

2. **Average Rank Similarity (ARS)**: This measure assesses the consistency in ranking the importance of the superpixels across different runs. A higher ARS score indicates that the importance ranks of the superpixels are more consistent across multiple runs than a lower ARS score. We use Rank Biased Overlap (RBO) score [21] to calculate the ARS across different runs for model 'Model' and explanation technique 'xp' as per the equation below.

$$ARS^{xp}_{Model} = \frac{\sum_{i=1}^{m-1}\sum_{j=i+1}^{m} rbo\_ext(\boldsymbol{R_i}, \boldsymbol{R_j})}{\binom{m}{2}} \tag{2}$$

Here, $\boldsymbol{R_i}$ and $\boldsymbol{R_j}$ are the ranked coefficient vectors from the i-th and j-th runs respectively, and $rbo\_ext(\boldsymbol{R_i}, \boldsymbol{R_j})$ represents the RBO (extrapolated) score between these two ranked coefficient vectors. We used the python package 'rbo' [3] to calculate the RBO scores, and we set the persistence parameter (p) = 0.2 to give more weightage to the top ranks. The term $\binom{m}{2}$ in the denominator of Eq. (2) is the number of unique pairs of models, which is used to average the rank similarities. ARS^{xp}_{Model} ranges between [0,1] where 1 denotes full match, and 0 indicates no match in the ranks of superpixels for any two runs.

3. **Combined Consistency Metric (CCM)**: The ASFE and ARS metrics quantify the sign entropy and the variance in the relative importance ranks of superpixels, respectively. However, we need a consolidated metric to understand and evaluate an XAI system. Hence, we combine both metrics to build our proposed metric, CCM, as:

$$CCM^{xp}_{Model} = (1 - ASFE^{xp}_{Model}) * ARS^{xp}_{Model} \tag{3}$$

CCM^{xp}_{Model} ranges between [0,1] where 0 denotes low consistency and 1 denotes full consistency in both sign entropy and superpixel importance ranks.

4. Experimental Setup

We conducted the experiments with two pre-trained image classification models - InceptionV3 [18] and ResNet50 [6] initialized with imagenet weights on Oxford-IIIT Pet Dataset [10] and Pascal VOC 2007 [4] datasets. We randomly select 50 images from each of the mentioned datasets and we analyze both selected models across 20 different runs. Our code was written in Python 3.8 and Tensorflow 2.6. In our experiments, we use a kernel size of (5,5) for Gaussian Blur. We have re-implemented LIME and used the code of BayLIME from the author's GitHub repository [24] for all our experiments. It was shown in [25] that BayLIME without priors behaves like LIME. We have selected BayLIME with Grad-CAM as prior for comparison with SLICE as it was shown to have better consistency and fidelity compared to LIME. We analyze SLICE, LIME, and BayLIME on each image and the selected pre-trained models. We then calculate the proposed consistency metric for each image-model pair for SLICE, LIME, and BayLIME (refer Sec. 5.1). Further, we use multiple methods to evaluate the fidelity of explanations from the ones mentioned

in Sec. 5.3. Further, we use the Wilcoxon rank test [20] to determine the statistical significance of our results and observations, owing to its non-parametric nature.

5. Results

5.1. Consistency Evaluation

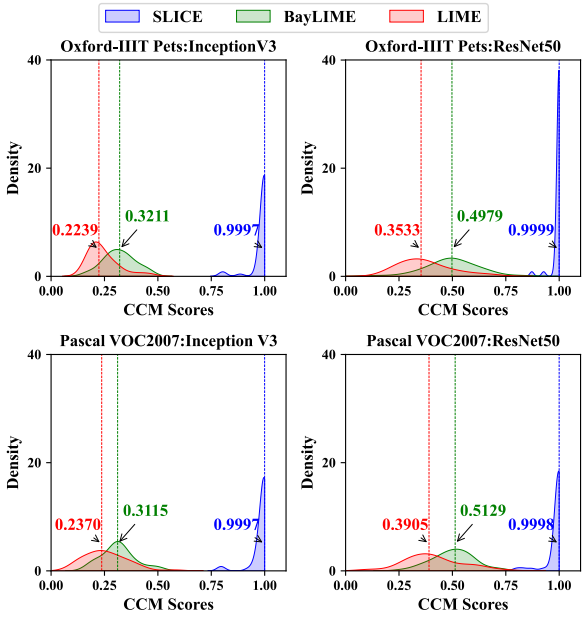


Figure 3. Distribution of CCM Scores for LIME, BayLIME and SLICE (higher is better)

Fig. 3 shows the distribution of the CCM scores for all the four combinations of datasets and models. The graph shows that BayLIME and LIME have much lower CCM scores than SLICE. Further, we conducted the Wilcoxon signed-rank test [20] to ascertain that the higher CCM scores of SLICE as compared to LIME and BayLIME are statistically significant.

The p-values from the tests were low, and the Test Statistics were high (See supplementary - Tab. S1). The notably low p-value and the substantially high value of the Test Statistic provide robust statistical evidence to reject the null hypothesis. This proves that the higher CCM scores of SLICE explanations than that of LIME and BayLIME are statistically significant making them significantly consistent.

5.2. Ablation Study for SLICE

SLICE has two major components, i.e., SEFE and Ada-Blur. We performed an ablation study to determine the contribution of both components separately. The details of the components are in Tab. 1. As seen from Fig. 3 and Fig. 4, SLICE_blur has a much higher CCM score than LIME but is slightly lower than SLICE. This is because using Ada-Blur as the perturbation technique, the perturbed images are created near to the original image. We observed this low variance of $\hat{Y}$ for the top predicted class for Ada-Blur vs. setting all pixels in the concerned superpixel to 0 as used in LIME and BayLIME (refer Fig. 2).

Table 1. Ablation settings with SLICE_blur, SLICE_FE, SLICE.

Method	SEFE	Ada-Blur
SLICE_blur	✗	✓
SLICE_FE	✓	✗
SLICE	✓	✓

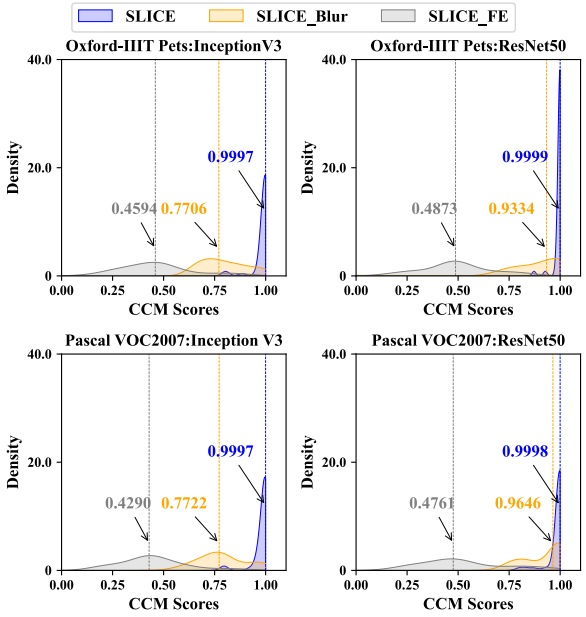


Figure 4. Distribution of CCM Scores for SLICE_blur, SLICE_FE and SLICE (higher is better)

Similarly, SLICE_FE has a higher CCM score than LIME but lower than SLICE_blur and SLICE. Without Ada-Blur, the perturbed images are created much further from the IE making it difficult for SLICE_FE to estimate the sign entropy. However, when we combine both approaches in SLICE, SEFE can correctly estimate the sign entropy of the superpixels and eliminate them. This is evident from the high CCM scores. Further, one of the components of CCM is the similarity of superpixel importance ranks across multiple runs. Thus the higher CCM score for SLICE proves that by eliminating superpixels with high sign entropy, we achieved significantly higher similarity in the relative importance ranks of superpixels, thereby contributing to the consistency of explanations.

Further, the noteably low p-values and high value of Test

Statistics of Wilcoxon rank test provide (details in Tab. S1) robust statistical evidence that the CCM scores of SLICE is higher than SLICE_blur and SLICE_FE.

5.3. Fidelity Evaluation of Explanations

5.3.1 Area Over Perturbation Curve

We use the Area Over Perturbation Curve (AOPC), a variation of insertion and deletion scores, to evaluate the fidelity of explanations for LIME, BayLIME, and SLICE. AOPC was proposed by [14] by extending the work of [1]. AOPC is used to analyze the drop in $\hat{Y}$ of an image by perturbing pixels(superpixels in our case) in the sequence of their importance. The original AOPC metric was defined for deletion, but we adapted it for insertion too. The adapted AOPC metric is defined below:

$$AOPC_M = \frac{1}{L+1}\left\langle \sum_{k=1}^{L} \Delta f(x,k) \right\rangle_{p(x)} \quad (4)$$

Where $\Delta f(x,k)$ represents the change in classifier output after k perturbation steps. For deletion of positive superpixels or insertion of negative superpixels, $\Delta f(x,k)$ is $f(x^{(0)}) - f(x^{(k)})$ and $x^{(0)}$ is the original image. For insertion of positive superpixels or deletion of negative superpixels, $\Delta f(x,k)$ is $f(x^{(k)}) - f(x^{(0)})$ and $x^{(0)}$ is the fully perturbed (blurred) image. $x^{(k)}$ is the input image after k perturbation steps for insertion, and for deletion, it is the blurred image after k steps of inserting the original image superpixels into the blurred image. L is the total number of perturbation steps. $\langle . \rangle_{p(x)}$ denotes the mean over all images in the dataset. M is the pixel deletion procedure, i.e., Most Relevant First (MoRF) or the Least Relevant First (LeRF) procedure. As we are using all the superpixels in our evaluation, the results for both pixel deletion/insertion procedures would be the same. Hence, we ran all our experiments with the MoRF procedure and will refer to it as AOPC. As the formulation of AOPC is based on the difference of probabilities of the initial image and the image obtained after insertion or deletion, a higher AOPC score for both insertion and deletion indicates higher fidelity. This is in contrast to the traditional insertion and deletion metrics, where a higher insertion Area Under the Curve (AUC) and a lower deletion AUC indicate higher fidelity.

We plot the Empirical Cumulative Distribution Function (ECDF) plots of the AOPC scores for deletion in Fig. 5a and for insertion in Fig. 5b. The AOPC scores for SLICE were higher than that of LIME and BayLIME. Further, it can be seen that there are few images for which the AOPC score was negative for LIME and BayLIME. A negative AOPC score would indicate that the resultant drop or increase in $\hat{Y}$ was opposite to what was expected by inserting or deleting a superpixel. As such, the explanations of LIME and BayLIME for those images have very low fidelity.

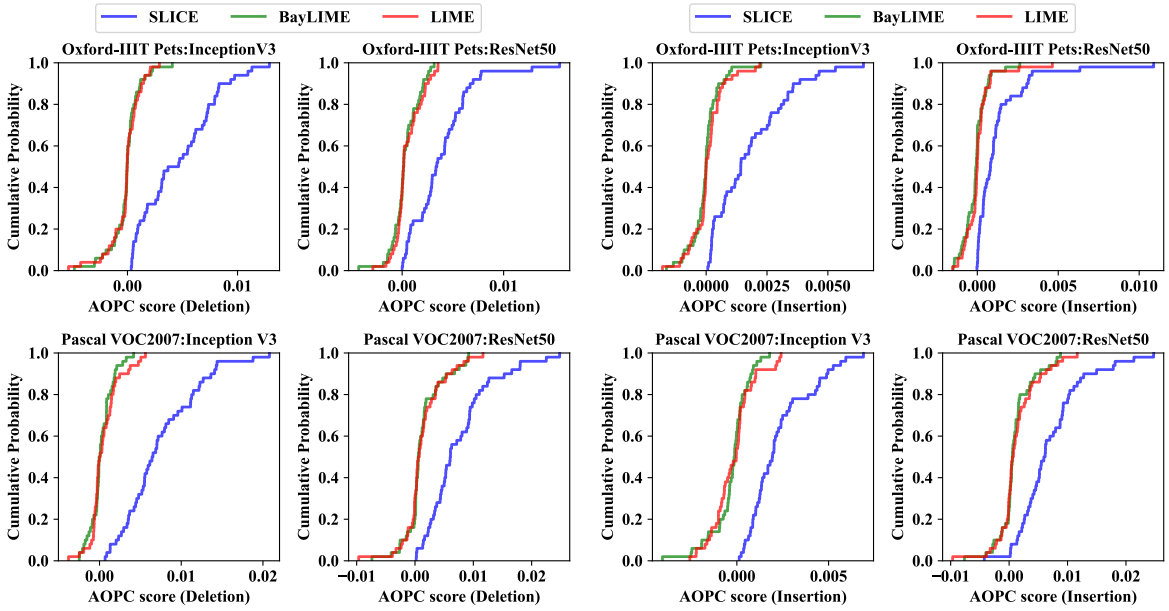
Table 2. Probability of observing a negative AOPC score denoted by Pr^- (Lower probability indicates higher fidelity) for different methods across different datasets and models.[O and P denote the Oxford-IIIT Pets and PASCAL VOC datasets, while I and R denote InceptionV3 and Resnet50 models.]

Method	O_I	O_R	P_I	P_R
$LIME^{ins}$	0.5687	0.5131	0.5283	0.3204
$BayLIME^{ins}$	0.5982	0.6227	0.6440	0.2288
$SLICE^{ins}$	**0.0**	**0.0545**	**0.0**	**0.0186**
$LIME^{del}$	0.4063	0.3304	0.3749	0.3204
$BayLIME^{del}$	0.4504	0.4468	0.5343	0.2251
$SLICE^{del}$	**0.0**	**0.0**	**0.0**	**0.0**

Table 3. Wilcoxon rank test results for comparison of LIME, BayLIME, and SLICE. AOPC(x,y) indicates the test where the null hypothesis H_0 was "The median of the differences $(AOPCscore(x) - AOPCscore(y))$ is equal to zero," and the alternative hypothesis was H_a was "The median of the differences $(AOPCscore(x) - AOPCscore(y))$ is greater than zero". [SLICE(S), LIME(L), and BayLIME(B); D:M denotes Dataset:Model; O refers to Oxford-IIT Pets and P refers to PASCAL VOC datasets. R denotes Resnet50 and I denotes Inception V3 models. W denotes the Test Statistic, M_Δ denotes the median of differences and Neg. Count denotes the number of negative differences (out of 50 images)]

Test	**D:M**	**W**	**p-value**	M_Δ	**Neg. Count**
Insertion					
AOPC(S,L)	O:I	1210	1.6e-08	0.001	2
AOPC(S,L)	O:R	1126	1.2e-06	0.001	3
AOPC(S,L)	P:I	1249	1.8e-09	0.002	1
AOPC(S,L)	P:R	1134	8.2e-07	0.006	9
AOPC(S,B)	O:I	1248	1.9e-09	0.002	1
AOPC(S,B)	O:R	1176	1.0e-07	0.001	4
AOPC(S,B)	P:I	1269	5.4e-10	0.003	0
AOPC(S,B)	P:R	1171	1.3e-07	0.01	8
Deletion					
AOPC(S,L)	O:I	1255	1.3e-09	0.01	1
AOPC(S,L)	O:R	1228	5.9e-09	0.003	4
AOPC(S,L)	P:I	1250	1.7e-09	0.01	3
AOPC(S,L)	P:R	1135	7.8e-07	0.01	11
AOPC(S,B)	O:I	1233	4.5e-09	0.005	3
AOPC(S,B)	O:R	1229	5.7e-09	0.003	3
AOPC(S,B)	P:I	1275	3.8e-10	0.006	0
AOPC(S,B)	P:R	1174	1.1e-07	0.004	5

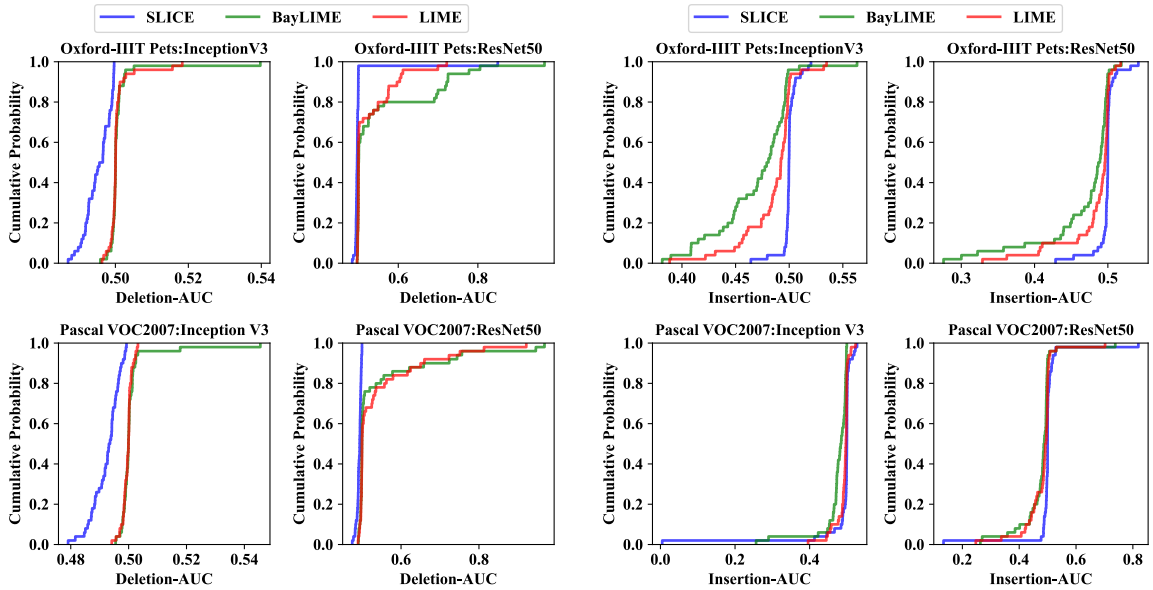
On the other hand, for SLICE, there was one image for which the AOPC score was lower than 0. Hence, we estimated the probability of observing a negative AOPC value (Pr^-) for an explanation from SLICE, LIME, and BayLIME. We used KDE to analyze the distribution of AOPC scores for explanations from the mentioned methods.



(a) ECDF plots of $AOPC$ deletion scores

(b) ECDF plots of $AOPC$ insertion scores

Figure 5. ECDF plots of AOPC (Higher $AOPC$ indicates higher fidelity)



(a) ECDF plots of Deletion AUC.(Lower is better)

(b) ECDF plots of Insertion AUC.(Higher is better)

Figure 6. ECDF plots of Deletion and Insertion AUC

We used Scott's rule of thumb (refer Sec. S4) to determine the bandwidth of KDE in our experiments. From Tab. 2, it can be seen that SLICE has significantly lower Pr^{-} as compared to LIME and BayLIME. This further proves the fidelity of SLICE explanations as compared to LIME and BayLIME.

The AOPC scores are low due to the nature of the perturbation used (Ada-Blur). We also performed Wilcoxon rank tests to ascertain that the higher AOPC values of SLICE explanations, as compared to those of LIME and BayLIME, were statistically significant. In our tests, the p-values were low, and the test statistics were high (Sec. 5.3.1). The re-

sults from our tests provide robust statistical evidence confirming that the AOPC values of SLICE explanations were significantly higher than those of LIME and BayLIME.

Table 4. Table presents Wilcoxon rank test results for comparing the Insertion and deletion AUCs of LIME, BayLIME, and SLICE. AUC(x,y) denotes a test with null hypothesis H_0 that the median difference in scores (insertion or deletion) between x and y is zero, against an alternative hypothesis H_a of a positive median difference. [SLICE (S), LIME (L), and BayLIME (B); D:M signifies Dataset:Model; with O for Oxford-IIT Pets, P for PASCAL VOC, R for Resnet50, and I for Inception V3. W represents the Test Statistic, M_Δ the median of differences, and Neg. Count the number of negatives in 50 images.]

Test	**D:M**	**W**	**p-value**	M_Δ	**Neg. Count**
Insertion					
AUC(S,L)	O:I	1124	1.3e-06	0.008	7
AUC(S,L)	O:R	962	8.7e-4	0.004	13
AUC(S,L)	P:I	852	1.9e-4	0.002	14
AUC(S,L)	P:R	955	1.1e-3	0.008	14
AUC(S,B)	O:I	1187	5.7e-08	0.019	4
AUC(S,B)	O:R	1138	6.8e-07	0.013	7
AUC(S,B)	P:I	1116	1.9-06	0.012	5
AUC(S,B)	P:R	1105	3.2e-06	0.015	6
Deletion					
AUC(L,S)	O:I	1240	3.0e-09	0.005	2
AUC(L,S)	O:R	1214	1.3e-08	0.005	2
AUC(L,S)	P:I	1266	6.5e-10	0.007	1
AUC(L,S)	P:R	1201	2.7e-08	0.01	6
AUC(B,S)	O:I	1234	4.3e-09	0.006	3
AUC(B,S)	O:R	1251	1.6e-09	0.006	2
AUC(B,S)	P:I	1274	4.0e-10	0.007	0
AUC(B,S)	P:R	1201	2.7e-08	0.008	5

5.3.2 Deletion and Insertion Game

We also performed the traditional insertion and deletion test [11] for LIME, BayLIME, and SLICE. Superpixels are added as per their importance in the insertion procedure, and the model's change of $\hat{Y}$ is noted. A higher fidelity explanation should have a higher area under the curve (AUC) for its insertion graph. Conversely, in the deletion procedure, superpixels are deleted, and the model's change in $\hat{Y}$ is noted. An explanation with higher fidelity should have a lower AUC for the deletion graph.

We plot the ECDF graphs for the Deletion and Insertion AUCs in Fig. 6a and Fig. 6b. The ECDF graph in Fig. 6a indicates that the AUCs, for the deletion procedure, of SLICE explanations was lower than that of LIME and BayLIME explanations. We performed the Wilcoxon rank tests on the AUCs obtained for all three methods to confirm this observation. In our tests, the p-values were extremely low, and the test statistics were high (Tab. 4).

The ECDF graph in Fig. 6b indicates that the AUCs for the insertion procedure of the explanations from SLICE were much higher than those from LIME and BayLIME. We performed Wilcoxon rank tests to ascertain that the higher insertion score of SLICE explanations, compared to those of LIME and BayLIME, were statistically significant. In our tests, the p-values were extremely low, and the test statistics were high (Tab. 4).

The results from our tests provide robust statistical evidence confirming that the explanations of SLICE were significantly superior in fidelity than those of LIME and BayLIME.

Table 5. Average running time of LIME, BayLIME, and SLICE (in seconds per image) for Inception V3 and Resnet50 models.

Method	**Inception_v3**	**Resnet50**
$LIME$	19.48s	11.73
$BayLIME$	19.70s	11.87
$SLICE$	88.94	44.73

5.4. Computation Time

Tab. 5 presents average runtime for LIME, BayLIME, and SLICE using Inception V3 and Resnet50 indicating the need to enhance SLICE's speed.

6. Limitations

[19] highlighted inconsistencies in saliency metrics due to unjustified perturbations. We address this with theoretically and empirically justified Gaussian Blur (Sec. 3.2). An extensive investigation of the reliability of fidelity metrics was not studied and can be a potential future work.

7. Conclusion and Future Work

The results of our experiments indicate that a holistic approach is more effective in stabilizing LIME explanations. Our proposed method, SLICE, used a novel feature elimination method and Gaussian blur with adaptive sigma selection as the perturbation technique, to stabilize explanations. Our feature elimination method, SEFE, adeptly estimates and discards superpixels with high sign variability. Gaussian blur with sigma selected adaptively, Ada-Blur, as perturbation technique constrained the perturbed samples to be closer to the IE and yet have an adequate variance for the surrogate model to learn. This led to a substantial reduction in the sign entropy and the variance of superpixels' importance, thus ensuring higher consistency. Further, our results also provide strong empirical evidence that the fidelity of SLICE explanations is significantly higher than that of LIME and BayLIME.

References

[1] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10 (7):e0130140, 2015. 6

[2] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE winter conference on applications of computer vision (WACV)*, pages 839–847. IEEE, 2018. 1

[3] Changyao Chen. Rank-biased overlap (rbo). https://pypi.org/project/rbo/, 2023. Accessed: 2023-08-08. 4

[4] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results, 2007. 4

[5] Alicja Gosiewska and Przemyslaw Biecek. Do not trust additive explanations. *arXiv preprint arXiv:1903.11420*, 2019. 1, 2

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4

[7] Gichan Lee and Scott Uk-Jin Lee. Towards reliable software analytics: Systematic integration of explanations from different model-agnostic techniques. *IEEE Software*, 2023. 1, 2

[8] Xuhong Li, Haoyi Xiong, Xingjian Li, Xiao Zhang, Ji Liu, Haiyan Jiang, Zeyu Chen, and Dejing Dou. G-lime: Statistical learning for local interpretations of deep neural networks using global priors. *Artificial Intelligence*, 314:103823, 2023. 1

[9] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017. 1

[10] Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and dogs. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012. 4

[11] Vitali Petsiuk, Abir Das, and Kate Saenko. Rise: Randomized input sampling for explanation of black-box models. *arXiv preprint arXiv:1806.07421*, 2018. 8

[12] Harish Guruprasad Ramaswamy et al. Ablation-cam: Visual explanations for deep convolutional network via gradient-free localization. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 983–991, 2020. 1

[13] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why should I trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016. 1

[14] Wojciech Samek, Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, and Klaus-Robert Müller. Evaluating the visualization of what a deep neural network has learned. *IEEE transactions on neural networks and learning systems*, 28(11):2660–2673, 2016. 6

[15] David W Scott. *Multivariate density estimation: theory, practice, and visualization*. John Wiley & Sons, 2015. 2, 4

[16] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017. 1

[17] Sharath M Shankaranarayana and Davor Runje. Alime: Autoencoder based approach for local interpretability. In *Intelligent Data Engineering and Automated Learning–IDEAL 2019: 20th International Conference, Manchester, UK, November 14–16, 2019, Proceedings, Part I 20*, pages 454–463. Springer, 2019. 2

[18] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 4

[19] Richard Tomsett, Dan Harborne, Supriyo Chakraborty, Prudhvi Gurram, and Alun Preece. Sanity checks for saliency metrics. In *Proceedings of the AAAI conference on artificial intelligence*, pages 6021–6029, 2020. 8

[20] Pauli Virtanen, Ralf Gommers, Travis E Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, et al. Scipy 1.0: fundamental algorithms for scientific computing in python. *Nature methods*, 17(3):261–272, 2020. 5

[21] William Webber, Alistair Moffat, and Justin Zobel. A similarity measure for indefinite rankings. *ACM Transactions on Information Systems (TOIS)*, 28(4):1–38, 2010. 4

[22] Muhammad Rehman Zafar and Naimul Mefraz Khan. Dlime: A deterministic local interpretable model-agnostic explanations approach for computer-aided diagnosis systems. *arXiv preprint arXiv:1906.10263*, 2019. 1, 2

[23] Yujia Zhang, Kuangyan Song, Yiming Sun, Sarah Tan, and Madeleine Udell. "Why Should You Trust My Explanation?" Understanding Uncertainty in LIME Explanations. *arXiv preprint arXiv:1904.12991*, 2019. 1

[24] Zhao. Baylime: A bayesian local interpretable model-agnostic explanation approach. https://github.com/x-y-zhao/BayLime, 2023. 4

[25] Xingyu Zhao, Wei Huang, Xiaowei Huang, Valentin Robu, and David Flynn. Baylime: Bayesian local interpretable model-agnostic explanations. In *Uncertainty in artificial intelligence*, pages 887–896. PMLR, 2021. 1, 2, 4

[26] Zhengze Zhou, Giles Hooker, and Fei Wang. S-lime: Stabilized-lime for model explanation. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pages 2429–2438, 2021. 1, 2