This WACV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

SSplain: Sparse and Smooth Explainer for Retinopathy of Prematurity Classification

Elifnur Sunger[1], Tales Imbiriba[2], Peter Campbell[3], Deniz Erdogmus[1], Stratis Ioannidis[1], Jennifer Dy[1]

[1]Northeastern University, Boston, MA, USA
[2]University of Massachusetts Boston, Boston, MA, USA
[3]Oregon Health & Science University, Portland, OR, USA

[1]{sunger.e, d.erdogmus, e.ioannidis, j.dy}@northeastern.edu, [2]tales.imbiriba@umb.edu,
[3]campbelp@ohsu.edu

Abstract

Neural networks are frequently used in medical diagnosis. However, due to their black-box nature, model explainers are used to help clinicians understand better and trust model outputs. This paper introduces an explainer method for classifying Retinopathy of Prematurity (ROP) from fundus images. Previous methods fail to generate explanations that preserve input image structures such as smoothness and sparsity. We introduce Sparse and Smooth Explainer (SSplain), a method that generates pixel-wise explanations while preserving image structures by enforcing smoothness and sparsity. This results in realistic explanations to enhance the understanding of the given black-box model. To achieve this goal, we define an optimization problem with combinatorial constraints and solve it using the Alternating Direction Method of Multipliers (ADMM). Experimental results show that SSplain outperforms commonly used explainers in terms of both post-hoc accuracy and smoothness analyses. Additionally, SSplain identifies features that are consistent with domain-understandable features that clinicians consider as discriminative factors for ROP. We also show SSplain's generalization by applying it to additional publicly available datasets. Code is available at https://github.com/neu-spiral/SSplain.

1. Introduction

Retinopathy of Prematurity (ROP) affects premature infants and is a leading cause of childhood blindness [15]. Early detection and treatment can prevent vision impairment; this has motivated the study of automated, machine-learning based methods for detecting the disease. Early work on ROP detection from fundus images [3, 40] used signal processing techniques to extract clinically relevant features; *dilation*, i.e., thickness of vessels in the retina, and *tortuosity*, as captured by vessel curvature. Generally, the more dilated and tortuous the vessels, the more severe the ROP [11].

While these approaches lead to interpretable algorithms, recent research has improved classification performance by employing convolutional neural network (CNN) architectures [6, 44] combined with attention-based mechanisms [49, 50]. For example, Brown *et al*. [6] propose a state-of-the-art ROP detection CNN based on the Inception architecture [38] that achieves 91% balanced accuracy. However, due to the black-box nature of neural networks, clinicians often hesitate to trust the diagnostic outcomes of such deep learning methods [17].

As an attempt to explain classifiers' decisions, the so-called black-box explainers [14, 30, 35] have been frequently used. When applied to image classifiers, they produce *explanation maps*, i.e., scores of pixels indicating their relevance to prediction.

Unfortunately, generic explainers fail to incorporate domain-specific image structures that might aid in interpretability. For example, fundus images, as shown in Figure 1(a), exhibit *sparsity*: images consist mostly of black pixels corresponding to the background, and discriminative features are concentrated on a subset of the vessels. Another relevant property is *smoothness*: regions of importance form contiguous segments of proximal vessel pixels of similarly (elevated) dilation and tortuosity. Black-box explainers for ROP detection need to consider and preserve the input image structure, such as smoothness and sparsity, to generate realistic explanation maps or attribution scores.

We introduce a Sparse and Smooth Explainer (SSplain) for ROP, which generates pixel-wise sparse and smooth explanations for the given black-box model and exploits the inherent image structure. Our contributions are as follows:

- We propose an explainability method that preserves inher-

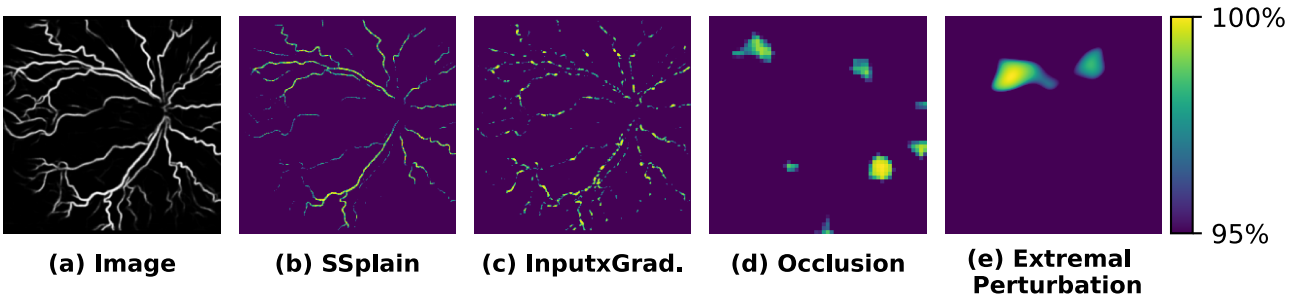


Figure 1. Explanation maps of (a) a sample image with ROP using (b) SSplain, (c) Input×Gradient [33], (d) Occlusion [52] and (e) Extremal Perturbation [13] methods. We visualize only the top 5% of pixels, as ranked by the corresponding explanation maps, coloring the remaining pixels as purple. SSplain generates sparse and contiguous regions of importance, that are also more dilated and tortuous than obscured vessels. Sparse method Input×Gradient focuses on similar areas but it is highly fragmented. Occlusion and Extremal Perturbation select discontinuous regions: tortuous vessels and dilated vessels, respectively. Both, however, fail to preserve vessel structure.

ent ROP image structures by enforcing explainer smoothness and sparsity. This approach enhances the understanding of the black-box predictor for classifying ROP images by highlighting relevant structures.

- This problem is non-convex and has combinatorial constraints; we use the Alternating Direction Method of Multipliers (ADMM) [5] to reduce this optimization to tractable subproblems.
- Through post-hoc performance and smoothness analyses, we demonstrate that SSplain outperforms several state-of-the-art competitors.
- We evaluate the explainers on two domain-specific metrics, dilation and tortuosity, which are discriminative factors for ROP. SSplain *consistently identifies more dilated and tortuous regions than competitors*, accentuating the superiority of SSplain towards ROP explainability. We stress that this occurs even though the tortuosity and dilation are neither explicitly measured nor incorporated in SSplain design objective.
- We perform a sanity check test [2] for explainability methods, showing that SSplain is sensitive to changes in the model, as desired.
- We apply SSplain to MNIST [22] and FMNIST [46], demonstrating its generalization and outperforming state-of-the-art methods in post-hoc analyses.

2. Related work

Numerous methods have been proposed to explain black-box models. They can be categorized into gradient-based and perturbation-based approaches. Gradient-based explainers [4, 32, 33, 35, 36] compute attribution scores using the gradients with respect to the inputs or intermediate feature maps. Perturbation-based approaches [13, 14, 30, 52] compute attribution scores via the neural network's response to perturbations. For instance, the Occlusion method [52] measures feature importance by analyzing changes in the model predictions when applying binary masks to image regions. Some gradient-based and perturbation-based explainability methods use attention weights to generate explanations [1, 8, 9]. These methods work only for Transformer-based black-box models and cannot explain the decisions of the state-of-the-art ROP classifier since it is CNN-based.

Recent work on the explainability of ROP classification includes the use of gradient-based explainers like DeepSHAP [43] and GradCAM [45]. However, these generic explainers do not integrate domain-specific image characteristics, such as sparsity and smoothness, which are important for ROP images.

Several explainability methods have attempted to incorporate sparsity or smoothness or both. The Input×Gradient [33] generates sparse explanations by multiplying the input with the model's output gradient with respect to the input. Nevertheless, it exhibits a lack of smoothness across neighboring vessels, as seen in Figure 1(c). Fong *et al*. [13, 14] define their objective function to generate smooth and sparse explanation masks by learning these masks through applying perturbations to images, such as blurring or replacing pixel values with a constant. In Fong *et al*. [14] the authors incorporate the ℓ_1 norm for sparsity, total variation norm, and upsampling from a low-resolution mask with the application of a Gaussian kernel for smoothness. While the ℓ_1 norm indeed leads to sparse solutions, it serves as an approximation of the ℓ_0 norm; therefore, it results in a relaxed sparsity constraint (see also our experiments with ℓ_1 constraints in Section 4). To leverage a hard sparsity constraint, Fong *et al*. [13] employ a ranking-based area loss and incorporate smoothness using a Gaussian kernel. However, starting from a priori non-smooth masks and smoothing them afterward leads to discontinuous regions and fails to preserve vessel structure, as seen in Figure 1(e).

There are multiple adaptations of the method proposed by Fong *et al*. [13] specifically for the medical domain [25, 42]. They focus on improving the perturbation method described by Fong *et al*. [13], which involves applying blurring or replacing pixel values with a constant value. However, they do not focus on improving the sparsity and smoothness of the explainer. In contrast, we propose a method that can incorporate sparsity constraints using the ℓ_0 or ℓ_1 norms, along with a smoothness constraint during the mask optimization process. Given the optimization problem with multiple constraints, including potentially non-convex constraints, we propose to solve this problem using the Alternating Direction Method of Multipliers (ADMM).

ADMM decomposes problems with combinatorial constraints into subproblems that can be solved efficiently [19, 53]. Recent studies [24, 39] have demonstrated that ADMM is also effective for solving non-convex problems. This has led to applications in various areas such as weight pruning [19, 26, 53], adversarial attacks [47], and ranking regression [48]; we expand this repertoire of applications.

3. Methodology

3.1. Problem formulation

Given a trained neural network model and an ROP image, our goal is to find pixels across the image that affect the model's decision significantly. One way of finding the importance of image pixels is by perturbing the image through masking and observing its effect on the model output.

Formally, a black-box model is defined as a function that maps input image or features to a probability distribution (using a softmax layer), denoted as $f : \mathbb{R}^{h\times w} \to \mathbb{R}^c, \mathbf{X} \mapsto f(\mathbf{X})$, where c is the number of classes. We consider an image mask $\mathbf{M} \in \mathbb{R}_+^{h\times w}$ to be applied to an input image $\mathbf{X} \in \mathbb{R}^{h\times w}$, where h and w denote the image height and width. We treat this mask as an explanation map: zero values correspond to non-discriminative pixels, while positive values rank remaining pixels with respect to their discriminative value. To exploit the vessel network structure of segmented fundus images, focusing on vessel features, we only consider non-zero mask elements over image vessels; formally, $\mathbf{M} \in S_0$ where $S_0 = \{\mathbf{M} : \texttt{supp}(M_{jk}) \subseteq \texttt{supp}(X_{jk})\}$, where $\texttt{supp}(\cdot)$ denotes the support (i.e., the non-zero pixels). Given such a mask, by taking the Hadamard product of $\mathbf{X}$ and $\mathbf{M}$, we obtain a *masked image* $\mathbf{X} \odot \mathbf{M} \in \mathbb{R}^{h\times w}$.

Our objective is to create masks $\mathbf{M}$ that highlight important, discriminative pixels across the image and suppress pixels that do not influence model decisions while satisfying two additional desiderata: ensuring mask *sparsity* and *smoothness*. As discussed in Section 1, sparsity allows us to focus only on important vessel pixels across the image, while smoothness enhances the mask segment contiguity. Given $\mathbf{X}$ and a corresponding ground truth label y, consider a loss function $l : \mathbb{R}^c \times \mathbb{R} \to \mathbb{R}, f(\mathbf{X}) \times y \mapsto l(f(\mathbf{X}), y)$. We find masks by solving the optimization problem in Eq. (1):

$$\begin{aligned} \min_{\mathbf{M}} \quad & l(f(\mathbf{X} \odot \mathbf{M}), y) + \lambda\nu(\mathbf{M}) \,. \\ \text{s.t.} \quad & \mathbf{M} \in S_1, \quad \mathbf{M} \in S_2 \end{aligned} \tag{1}$$

Here, the constraint S_1 enforces sparsity, S_2 enforces that mask values stay within the range $[0, 1]$, while the regularization $\nu(\cdot)$ enforces smoothness, and $\lambda \geq 0$ is a regularization parameter. In particular, the sparsity constraint is defined as $S_1 = \{\mathbf{M} : \|\mathbf{M}\|_p \leq \alpha_p\} \cap S_0$ to limit the number of nonzero pixels, where $\|\cdot\|_p$ denotes the ℓ_p norm where $p \in \{0, 1\}$, i.e., the number of nonzero elements if $p = 0$ and the sum of the absolute values of elements if $p = 1$. Here, $\alpha_0 \in \mathbb{N}$ and $\alpha_1 \in \mathbb{R}^+$ are constants. Constraint S_2 is defined as $S_2 = \{\mathbf{M} : M_{jk} \in [0, 1]\} \cap S_0$. Finally, we enforce smoothness using the total variation function [7, 18]: $\nu(\mathbf{M}) = \sum_i \|\mathbf{M}_{i\cdot} - \mathbf{M}_{i-1\cdot}\|_1 + \sum_j \|\mathbf{M}_{\cdot j} - \mathbf{M}_{\cdot j-1}\|_1$, where $\mathbf{M}_{i\cdot}$ and $\mathbf{M}_{\cdot j}$ denote the rows and columns of $\mathbf{M}$.

We follow a *supervised approach* with label y available. However, an unsupervised scenario is possible by using the most likely output class as an objective, or taking the expectation of the loss with respect to the class posterior [10, 51].

3.2. Computing masks via ADMM

The optimization problem in Eq. (1) is non-convex with combinatorial constraints. We solve it using the Alternating Direction Method of Multipliers (ADMM), since it allows to decompose the problem into three subproblems that can be solved efficiently and separately [19, 53]. For specific non-convex constraints such as the ℓ_0 norm, the ADMM process carries out exactly as the convex case [5]. Fong et. al. [14] have a similar problem definition where they use ℓ_1 norm for sparsity. Nevertheless, ℓ_1 norm serves as an approximation of the ℓ_0 norm. For the weight-pruning problem, Zhang *et al*. [53] demonstrate that employing the ℓ_0 norm with the ADMM process results in sparser solutions compared to ℓ_1 norm-based methods. Using ADMM allows for the use of either ℓ_0 or ℓ_1 norms for sparsity constraints, whereas other methods cannot use the ℓ_0 norm.

We rewrite Eq. (1) with auxiliary variables $\mathbf{M}_1$ and $\mathbf{M}_2$ in Eq. (2), where $g_{S_i}(\xi)$ represent the indicator functions such that $g_{S_i}(\xi) = 0$ if $\xi \in S_i$ and $g_{S_i}(\xi) = +\infty$, otherwise.

$$\begin{aligned} \min_{\mathbf{M}} \; & l(f(\mathbf{X}\odot\mathbf{M}),y) + \lambda\nu(\mathbf{M}) + g_{S_1}(\mathbf{M}_1) + g_{S_2}(\mathbf{M}_2). \\ \text{s.t.} \; & \mathbf{M} = \mathbf{M}_1, \mathbf{M} = \mathbf{M}_2 \end{aligned} \tag{2}$$

Defining dual variables $\mathbf{U}_1, \mathbf{U}_2 \in \mathbb{R}^{h\times w}$ and the penalty value ρ, we write the augmented Lagrangian problem as:

$$\begin{aligned} &\mathcal{L}(\mathbf{M}, \mathbf{M}_1, \mathbf{M}_2, \mathbf{U}_1, \mathbf{U}_2) = l(f(\mathbf{X}\odot\mathbf{M}), y) + \lambda\nu(\mathbf{M}) + \\ &\sum_{i=1}^{2} \Big[g_{S_i}(\mathbf{M}_i) + \rho(\mathrm{Tr}(\mathbf{U}_i^T(\mathbf{M} - \mathbf{M}_i))) + \frac{\rho}{2}(\|\mathbf{M} - \mathbf{M}_i\|_2^2) \Big]. \end{aligned} \tag{3}$$

ADMM algorithm iteratively optimizes the augmented Lagrangian via a primal-dual approach until convergence or a maximum number of iterations. At iteration k, the steps are:

$$\mathbf{M}^{(k)} := \arg\min_{\mathbf{M}} \mathcal{L}(\mathbf{M}, \mathbf{M}_1^{(k-1)}, \mathbf{M}_2^{(k-1)}, \mathbf{U}_1^{(k-1)}, \mathbf{U}_2^{(k-1)}), \tag{4a}$$

$$\mathbf{M}_1^{(k)}, \mathbf{M}_2^{(k)} := \arg\min_{\mathbf{M}_1, \mathbf{M}_2} \mathcal{L}(\mathbf{M}^{(k)}, \mathbf{M}_1, \mathbf{M}_2, \mathbf{U}_1^{(k-1)}, \mathbf{U}_2^{(k-1)}), \tag{4b}$$

$$\mathbf{U}_i^{(k)} := \mathbf{U}_i^{(k-1)} + \rho(\mathbf{M}^{(k)} - \mathbf{M}_i^{(k)}) \text{ for } i = 1, 2. \tag{4c}$$

The problem in Eq. (4a) is equivalent to Eq. (5) and can be solved using standard gradient descent.

$$\mathbf{M}^{(k)} := \arg\min_{\mathbf{M}} l(f(\mathbf{X} \odot \mathbf{M}), y) + \lambda\nu(\mathbf{M}) + \sum_{i=1}^{2} \frac{\rho}{2} \|\mathbf{M} - \mathbf{M}_i^{(k-1)} + \mathbf{U}_i^{(k-1)}\|_2^2 . \tag{5}$$

Eq. (4b) for $\mathbf{M}_1$ and $\mathbf{M}_2$ is equivalent to:

$$\mathbf{M}_i^{(k)} := \arg\min_{\mathbf{M}_i} g_{S_i}(\mathbf{M}_i) + \frac{\rho}{2} \|\mathbf{M}^{(k)} - \mathbf{M}_i + \mathbf{U}_i{}^{(k-1)}\|_2^2 . \tag{6}$$

Crucially, the problems in Eqs. (6) correspond to Euclidean projection operators that can be solved efficiently in polynomial time [5, 53]. Therefore, we define projections for three cases: S_1 with the ℓ_0 norm, S_1 with the ℓ_1 norm, and S_2. In particular, we define the Euclidean projection operator as: $\Pi_{S_i} = \arg\min_{\mathbf{M}_i \in S_i} \|(\mathbf{M}^{(k)} + \mathbf{U}_i{}^{(k-1)}) - \mathbf{M}_i\|_2^2$, which maps $(\mathbf{M}^{(k)} + \mathbf{U}_i{}^{(k-1)})$ onto the set S_i. When $S_1 = \{\mathbf{M} : \|\mathbf{M}\|_0 \leq \alpha_0\} \cap S_0$, where $\alpha_0 \in \mathbb{N}$ is a constant, we update $\mathbf{M}_1{}^{(k)}$ by keeping the top α_0 values of $(\mathbf{M}^{(k)} + \mathbf{U}_1^{(k-1)})$ and setting the rest to zero. We can perform this despite the non-convexity of the set S_1 with the ℓ_0 norm. For $S_1 = \{\mathbf{M} : \|\mathbf{M}\|_1 \leq \alpha_1\} \cap S_0$, where $\alpha_1 \in \mathbb{R}^+$ is a constant, we follow the projection operation described by Duchi *et al*. [12]. Lastly, for S_2, we update $\mathbf{M}_2{}^{(k)}$ by setting the values of $(\mathbf{M}^{(k)} + \mathbf{U}_2^{(k-1)})$ that are smaller than zero to zero and larger than one to one. Algorithm 1 outlines the ADMM process for computing masks.

Algorithm 1 SSplain Algorithm

Input: Black-box model $f(\cdot)$, Input image $\mathbf{X}$, Target label y, Loss function $l(\cdot)$, Sparsity parameter α, Regularization parameter λ, Penalty value ρ, Maximum iteration K
Output: Attribution Mask $\mathbf{M}$

Initialize $\mathbf{M}^0$
$\mathbf{U}_i^0 \leftarrow \mathbf{0}$ for $i = 1, 2$
$\mathbf{M}_i^0 = \Pi_{S_i}(\mathbf{M}^0 + \mathbf{U}_i{}^0)$ for $i = 1, 2$
for k $\leftarrow 1, \ldots,$ K **do**
  $\mathbf{M}^{(k)} \leftarrow \arg\min_{\mathbf{M}} \big[l(f(\mathbf{X} \odot \mathbf{M}), y) + \lambda\nu(\mathbf{M}) + \frac{\rho}{2} \sum_{i=1}^{2} \|\mathbf{M} - \mathbf{M}_i^{(k-1)} + \mathbf{U}_i^{(k-1)}\|_2^2 \big]$
  $\mathbf{M}_i^{(k)} = \Pi_{S_i}(\mathbf{M}^{(k)} + \mathbf{U}_i{}^{(k-1)})$ for $i = 1, 2$
  $\mathbf{U}_i^{(k)} \leftarrow \mathbf{U}_i^{(k-1)} + \rho(\mathbf{M}^{(k)} - \mathbf{M}_i^{(k)})$ for $i = 1, 2$
end for
return $\mathbf{M}^{(K)}$

4. Experiments

4.1. Dataset and Model Details

ROP Dataset The ROP dataset [6] contains 224×224 segmented fundus images. A committee of three experts assigned Reference Standard Diagnostic (RSD) labels [31] as "Plus", "Pre-plus", or "Normal", indicating severe, intermediate, or no ROP, respectively. For training a classifier, we followed the setup by Brown *et al*. [6]. We split the data into training and testing sets with 5561 and 100 images, respectively. We use this dataset since a state-of-the-art classification method exists for it, the ROP classification setup proposed by Brown *et al*. [6], which achieves a balanced accuracy of 91%. Following Brown *et al*. [6], we train a classifier leveraging the Inception architecture [38], while reserving the test set for evaluating different explainers. Note that ROP images in this work are segmented fundus images, which is important since disease severity is determined by the appearance of retinal vessels [11].

The focus of our work is not model training. To evaluate SSplain on the ROP dataset, we need a trained model, which we get from the ROP classifier by Brown *et al*. [6].

We initialize masks with values proportional to the corresponding pixel intensity between 0 and 1, as follows: $\mathbf{M} = \frac{\mathbf{X}}{\|\mathbf{X}\|_\infty}$. The mask initialization depends on the use case. Alternatively, one can initialize masks as all ones assigned to the support of the image using S_0, where $M_{j,k}^0 = 1$ for $M_{j,k} \in S_0$ with $S_0 = \{\mathbf{M} : \texttt{supp}(M_{jk}) \subseteq \texttt{supp}(X_{jk})\}$. $\texttt{supp}(\cdot)$ denotes the support (i.e., the non-zero pixels). See Appendix I for mask initialization comparison. Intensity-based initialization is appropriate for our use case, as the optimization problem is constrained by sparsity and smoothness, and vessel intensity is an appropriate starting point.

We use cross-entropy for $l(\cdot, \cdot)$ and run 50 iterations of ADMM for each image. We utilize Adam optimizer [20] to solve problem (4a) with 0.01 learning rate, $\rho = 0.01$ and $\lambda = 10^{-5}$. We set α_0 in S_1 to 50% of $\|\mathbf{X}\|_0$ for the ℓ_0 norm. To maintain the same sparsity constraints in both the ℓ_0 and ℓ_1 norm cases, we use α_0 to determine α_1. Specifically, when using the ℓ_1 norm in S_1, where $S_1 = \{\mathbf{M} : \|\mathbf{M}\|_1 \leq \alpha_1\} \cap S_0$, we set α_1 based on α_0. At each ADMM iteration, α_1 is set to the sum of the top α_0 values of $(\mathbf{M}^{(k)} + \mathbf{U}_1^{(k-1)})$. See Appendix C for analyses with different α levels.

Additional Datasets We apply SSplain to two other datasets, MNIST [22] and FMNIST [46], where images exhibit sparse and smooth structures similar to ROP images. This allows us to evaluate the generality of SSplain.

We train the LeNet-5 [23] separately on the MNIST and FMNIST datasets using the Adam optimizer [20] with learning rate 0.001, weight decay 0.0001, for 50 epochs, batch size 32, and early stopping. This results in two separate black-box models, achieving 98.53% test accuracy on MNIST and 88.67% on FMNIST. We then perform our post-hoc explainability analyses on 500 images from the test sets. Refer to Appendix E for additional details on the MNIST [22] and FMNIST [46] datasets and model details.

4.2. Competing Methods

We compare SSplain with nine state-of-the-art explainers available within TorchRay [13] and Captum [21] libraries. Here, we provide the application details for the ROP dataset. *Gradient-based:* Saliency [35], Guided Grad-CAM [32], Integrated Gradients [36] and DeepSHAP [27]. We apply Guided Grad-CAM method to the output of the last Inception layer ("inception 5b"). *Perturbation-based:* KernelSHAP [27], LIME [30] and Occlusion [52]. LIME and KernelSHAP employ 200 input samples. Occlusion uses 16×16 windows with 4×4 strides. *Incorporating sparsity or smoothness or both:* Extremal Perturbation [13] and Input$\times$ Gradient [33]. Extremal Perturbation uses a Gaussian blur, a "hybrid" loss with 400 iterations, 0.05 smoothing factor, and a target area to 50% of $\|\mathbf{X}\|_0$. It initializes mask values to all ones. See Appendix D for method details, and Appendix E for the application to MNIST and FMNIST.

4.3. Experimental Setup

Insertion and Deletion Tests We follow a supervised approach to generate explanations, using true labels as targets. This is because we evaluate explainer performance not only based on accuracy but also on domain-specific feature similarity for ROP, which are disease and severity indicators. For consistency, we use this approach throughout the paper. See Appendix J for results using an unsupervised setup, using the model predictions as targets. Subsequently, we evaluate several metrics defined in Section 4.4 using the insertion and deletion processes described by Petsiuk *et al*. [29]. The insertion/deletion process evaluates performance by adding/ removing image pixels with the highest attribution scores. We evaluate this process with respect to the sparsity level s, representing the fraction of pixels selected between 0% and 100%. At each s, we obtain a corresponding image $\mathbf{X}_s$. Some methods pick black pixels, i.e., the background. Therefore, we also evaluate performance with respect to the normalized sparsity $\kappa_s = \frac{\|\mathbf{X}_s\|_1}{\|\mathbf{X}\|_1}$.

Sanity Checks We evaluate the sensitivity of SSplain using "Model Parameter Randomization Test" by Adebayo *et al*. [2]. We recursively randomize the model weights, layer by layer from top to bottom, while the explainer generates attribution scores for each case. This test assesses explainer sensitivity to model changes, and explainers should produce different attribution scores with randomized weights.

Computing Infrastructure We ran experiments using PyTorch [28] on an Intel Core i7-9700K CPU with 64GB RAM.

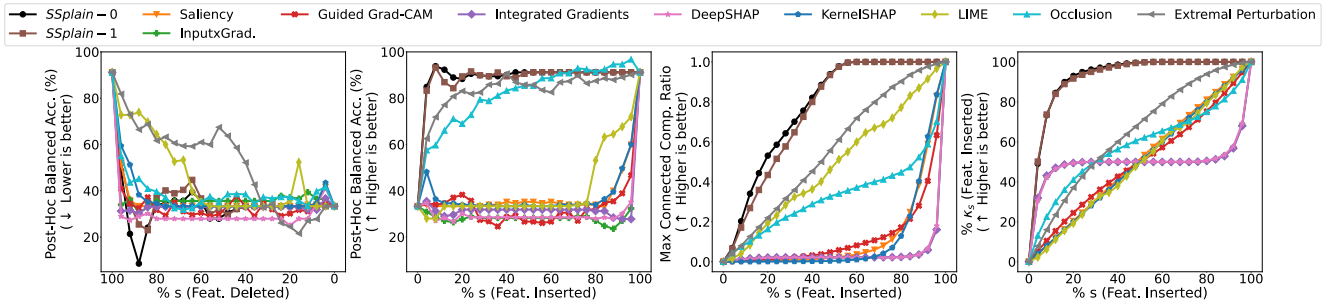
Computational Costs SSplain's execution time depends on the number of ADMM iterations, image size, and size of the black-box model. At each iteration, we perform a standard gradient descent (line 5 of Algorithm 1) and projection operations (line 6 of Algorithm 1). These projection operations are polynomial in time with respect to the image size, where we select the top α values for the ℓ_0 norm projection (S_1 constraint) and apply clipping for the S_2 constraint.

SSplain's runtime is comparable with the perturbation methods, the explainability family to which it belongs. Extremal Perturbation's runtime depends on the iteration number, image size, patch size, and model size. KernelSHAP and LIME depend on the model size and the perturbation samples. Occlusion depends on the model size, image size, patch size, and stride. Gradient-based methods depend on model size and are typically faster than input-perturbation methods. Note that explanations and runtimes are sensitive to method configurations. Please see Appendix H for the execution times.

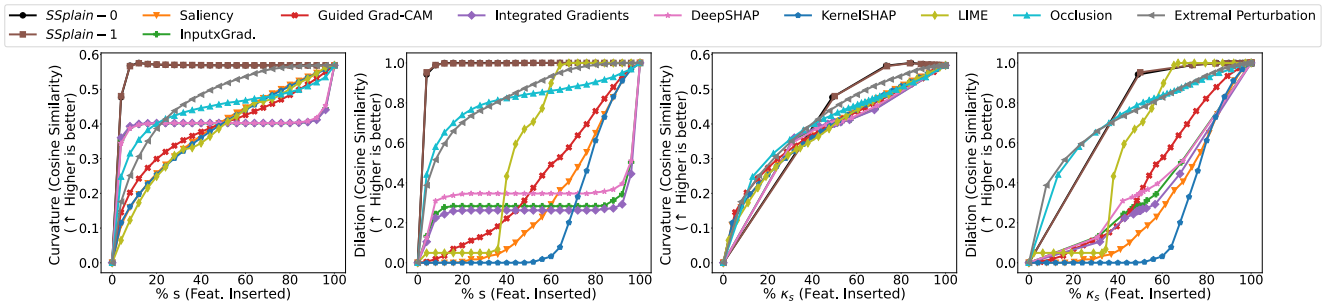
4.4. Evaluation Metrics

We evaluate explainers' performance on balanced accuracy. We use "Connected Components" to evaluate smoothness on ROP data. We also define two domain-related metrics to measure the explainer's ability to capture semantically meaningful ROP features: *tortuosity* and *dilation*.

Connected components We measure the smoothness of explanation maps, focusing particularly on the smoothness of vessel structures at each insertion step s. To achieve this, we extract graphs from $\mathbf{X}$ and $\mathbf{X}_s$, and identify the length of the largest connected component in each graph. This length is the number of nodes in the largest connected component. To both extract graphs and find the connected components in graphs, we use the NetworkX library [16]; we describe processes in Appendix A.1. We compute the ratio of the length of the largest connected component in $\mathbf{X}_s$ to the length of the largest connected component in $\mathbf{X}$. A higher largest connected component value at s indicates that the inserted pixels create smoother vessel structures.



(a) Accuracy, connected components, normalized sparsity κ_s metrics vs. sparsity s.



(b) Curvature and dilation analyses vs. sparsity s and normalized sparsity κ_s.

Figure 2. Comparison of explainers for the ROP dataset: SSplain-0 (S_1 with ℓ_0 constraint), SSplain-1 (S_1 with ℓ_1 constraint), Saliency [35], Input×Gradient [33], Guided Grad-CAM [32], Integrated Gradients [36], DeepSHAP [27], KernelSHAP [27], LIME [30], Occlusion [52] and Extremal Perturbation [13]. (a) We report the average of: Post-hoc balanced accuracy with deletion (lower is better) and insertion (higher is better) of pixels with the highest attribution scores, connected components ratio during the insertion process, and sparsity s versus normalized sparsity κ_s during the insertion process. (b) Curvature and dilation similarity with insertion process with respect to sparsity s and normalized sparsity κ_s. In (a), both SSplain methods consistently outperform competitors and competitors give importance to the background. In (b), both at the same s and κ_s level, SSplain methods exhibit higher curvature and dilation similarity.

Curvature Tortuosity is an ROP indicator. Therefore, we also evaluate the explainers' ability to capture tortuous regions at each insertion step s. For this, we measure the similarity of images at each insertion step s with pixel-wise curvature values $\mathbf{C} \in \mathbb{R}_{+}^{h \times w}$ for $\mathbf{X}$. These curvature values are calculated using the Tubular Curvature Filter Method [37]. Higher pixel-wise curvature values indicate greater tortuosity. Refer to the Appendix A.2 for details on this metric.

We use $\cos(\mathbf{X}_s, \mathbf{C}) = \frac{\mathbf{X}_s \cdot \mathbf{C}}{\|\mathbf{X}_s\| \|\mathbf{C}\|}$ to compute the cosine similarity between the curvature values $\mathbf{C}$ and $\mathbf{X}_s$. A higher value at s shows that the inserted pixels capture tortuous regions, and doing this at an earlier s indicates that the explainer assigns greater importance to these regions.

Dilation Another disease indicator is dilation, i.e., dilated vessels on the retina. To assess if explainers identify dilated vessels, we measure the similarity between dilated vessels in $\mathbf{X}$ and $\mathbf{X}_s$ during the insertion process. We extract dilation using the Average Segment Diameter (ASD) [3]. ASD measures the total number of pixels in a vessel branch divided by the curve length, allowing us to evaluate how well the explainer captures dilation at each insertion step s.

We evaluate $\cos(\mathbf{D_s}, \mathbf{D})$, where $\mathbf{D} \in \mathbb{R}^{b \times 1}$ contains the ASD features from $\mathbf{X}$ and $\mathbf{D}_s$ from $\mathbf{X}_s$. b is the number of vessel branches. Similar to the curvature metric, a higher value at an earlier s indicates that the explainer assigns greater importance to dilated vessel regions. We extract ASD features following Ataer *et al.* [3] and Tian *et al.* [41]. See Appendix A.3 for details.

4.5. Post-hoc and Similarity Analyses

We report averages of the metrics defined above over the test set for ROP dataset. SSplain-0 refers to when S_1 includes the ℓ_0 norm as the sparsity constraint, while SSplain-1 refers to when it includes the ℓ_1 norm. Figure 2 shows the evaluation of explainability methods. Across all analyses from Figure 13a, SSplain with both ℓ_0 and ℓ_1 consistently outperforms comparative methods, demonstrating lower deletion and higher insertion post-hoc accuracy, as well as higher values for max connected components ratio. Across all analyses, SSplain-0 performs as well as or better than SSplain-1. Thus, we include only SSplain-0 in the following analyses.

Except for the deletion analysis, the third-best is the Extremal Perturbation. The last column of Figure 13a shows competitors assign importance to the background, requiring

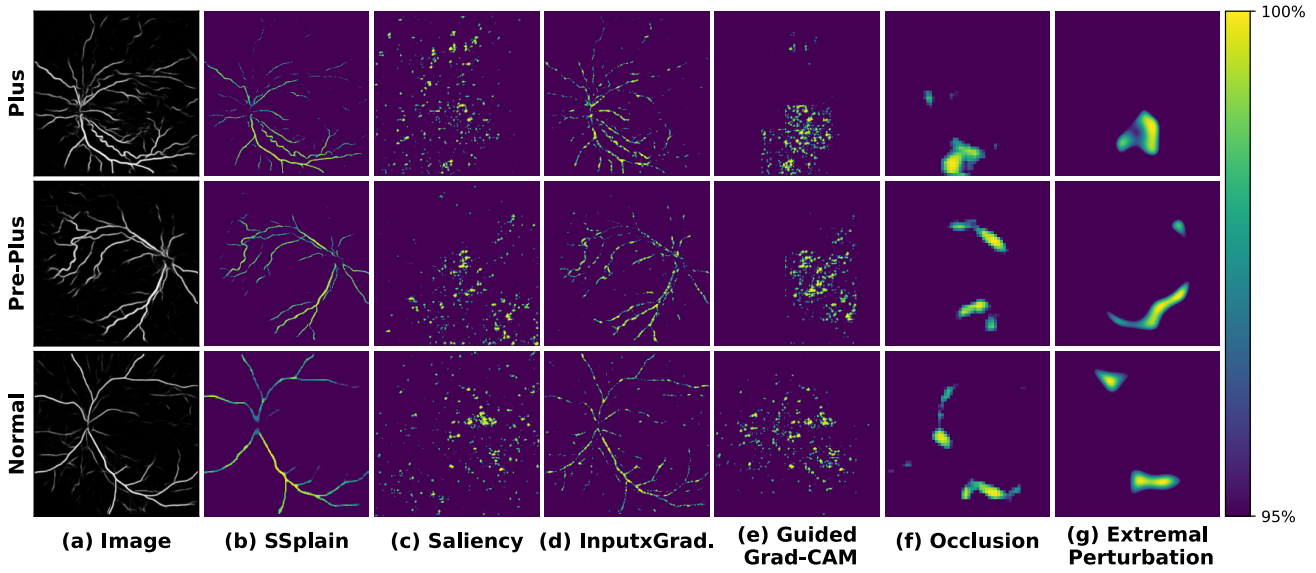


Figure 3. Comparison with other explainers for the ROP dataset. From left to right: (a) images, explanation maps generated using (b) SSplain-0, (c) Saliency [35], (d) Input×Gradient [33], (e) Guided Grad-CAM [32], (f) Occlusion [52] and (g) Extremal Perturbation [13]. We visualize only the top 5% of pixels, as ranked by the corresponding explanation maps, coloring the remaining pixels as purple. SSplain effectively preserves image structures and assigns more importance to tortuous and dilated regions. Except Input×Gradient, competitors treat images as a whole and assign importance to pixels that are not associated with vessels.

more steps to reach the κ_s level that SSplain achieves faster. Figure 13b shows that at the same s and κ_s level, *SSplain exhibits higher curvature and dilation similarity*, indicating better capture of domain-related features. See Appendix B for an ablation study on the ROP dataset.

4.6. Visualization

Visualizing the most important features can be challenging when comparing the outputs of the different methodologies and objectives (see Appendix F.2). To circumvent this issue, we visualize only the top 5% of pixels, as ranked by the corresponding explanation maps, coloring the remaining pixels purple. We illustrate this visualization procedure for the examples from the ROP dataset in Figure 8. Here, we only include SSplain with the ℓ_0 norm used in S_1. SSplain preserves image structures and assigns more importance to pixels with high tortuosity and dilation while maintaining smoothness. The S_0 and S_1 constraints in SSplain's optimization provide this sparsity advantage. Among competitors, only Input×Gradient has an S_0-like component, restricting explanations by multiplying the input with the output gradient. While Input×Gradient also preserves vessel shapes, it fails to generate smooth attribution scores and capture tortuous and dilated areas. SSplain generates attribution scores specifically for pixels associated with vessels by preserving their structure and enforcing sparsity. In contrast, competitors treat images as a whole and thus do not distinctly focus on the pixels associated with vessels. See

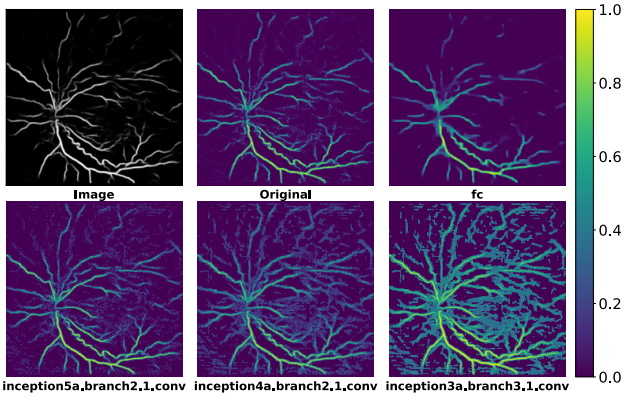
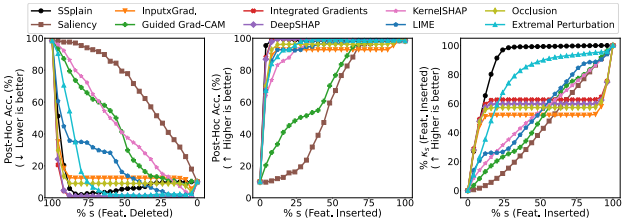
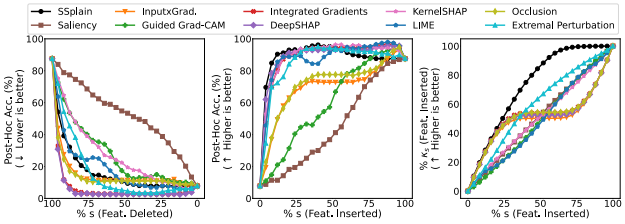


Figure 4. Sanity check [2] for an ROP image. Model weights are progressively randomized from the "fc" layer to the "inception3a.branch3.1.conv" layer in the Inception model [38]. The first row shows a sample image, original attribution scores, and attribution scores with only the "fc" layer randomized. The second row provides examples from the other layers during progressive randomization. SSplain is sensitive to model weights, i.e., attribution scores change when we randomize the weights as desired. S_0 ensures the mask applies only to the vessels, so the mask values for the background do not change.

Appendix G.1 for visualizations of SSplain-0 and SSplain-1. Refer to Appendix G.2 for SSplain-0 and Extremal Perturbations (pixel-wise), which uses the ℓ_1 norm to illustrate



(a) MNIST results.



(b) FMNIST results.

Figure 5. Comparison of explainers: SSplain-0, Saliency [35], Input×Gradient [33], Guided Grad-CAM [32], Integrated Gradients [36], DeepSHAP [27], KernelSHAP [27], LIME [30], Occlusion [52] and Extremal Perturbation [13] for (a) MNIST and (b) FMNIST datasets. From left to right, we report the average of: Post-hoc accuracy with deletion (lower is better) and insertion (higher is better) of pixels with the highest attribution scores, and sparsity s vs normalized sparsity κ_s during the insertion process. For MNIST, SSplain outperforms the competing methods in all analyses. It also stabilizes earlier than other methods, meaning that performance changes become minimal when inserting or deleting pixels in later attempts. For FMNIST, we observe that SSplain performs best in accuracy analysis with the insertion process and in normalized sparsity κ_s analysis. Input×Gradient and Occlusion outperform SSplain in deletion analyses. Despite their superior performance in deletion analyses, these methods significantly underperform compared to SSplain in insertion analysis, having lower insertion accuracy.

an example where both methods share an all-ones mask initialization and SSplain is without S_0.

Sanity Check In Figure 4, we visualize the sensitivity of the attribution scores by progressively randomizing the model weights from top to bottom, as proposed by Adebayo *et al*. [2], and apply SSplain-0 to the ROP dataset. This visualization shows how sensitive attribution scores are to changes in model weights. It is desired that the scores vary with randomized weights. We observe these changes in Figure 4 that the attribution scores change with the randomized weights. Note that S_0 ensures the mask applies only to the vessels, so background mask values do not change.

4.7. Analyses on the Additional Datasets

MNIST Figure 5a demonstrates the performance of explainability methods across different analyses and metrics for the MNIST dataset. SSplain-0 consistently outperforms the competing methods, demonstrating lower deletion and higher insertion post-hoc accuracy, as well as higher normalized sparsity κ_s. Additionally, SSplain stabilizes early, reaching its maximum sooner and maintaining stability until the end of the insertion process. Similar to ROP results, competitors prioritize both background and non-zero pixels, requiring more steps to achieve the κ_s level that SSplain reaches in just a few steps. Refer to Appendix F.2 for attribution score visualizations of the MNIST dataset.

FMNIST Figure 5b shows the performance of explainers across different analyses and metrics for the FMNIST dataset. Although SSplain-0 excels in insertion-based accuracy analysis and in normalized sparsity κ_s analysis, other methods such as Occlusion and Input×Gradient perform better than SSplain in deletion analysis. These methods, however, significantly underperform compared to SSplain in insertion analyses, exhibiting lower insertion accuracy. This means that the pixels with the highest attribution scores in the Occlusion and Input×Gradient methods significantly affect the classification when removed from the image but do not impact it when added. Similar to the MNIST and ROP results, SSplain stabilizes earlier. See Appendix F.2 for attribution score visualizations of the FMNIST dataset.

5. Conclusion

This paper proposes a black-box explainer designed to preserve input image structures by enhancing smoothness and sparsity for ROP classification. Unlike previous methods, SSplain effectively maintains input image structures, such as smoothness and sparsity, thus it generates realistic explanations to enhance clinicians' trust in model outputs. To generate attribution scores while ensuring both sparsity and smoothness, we define and solve a non-convex optimization problem with combinatorial constraints using ADMM. Experiments demonstrate that SSplain outperforms nine commonly used explainers not only in post-hoc performance but also in domain-based feature similarity analyses of tortuosity and dilation. Additionally, the sanity check visualizations show that SSplain is sensitive to changes in the model, which is desired for explainers.

In addition to showing SSplain's performance on the ROP dataset, our experiments include additional results on the MNIST and FMNIST datasets to demonstrate SSplain's applicability to other domains. SSplain is specifically defined for segmented ROP images, which have sparse and smooth characteristics. Therefore, it is also dependent on the segmentation quality in terms of smoothness and sparsity. While we define the optimization problem specifically for this, for other possible problem formulations requiring different optimization constraints, one can add or substitute constraints based on domain knowledge and solve them using ADMM.

Acknowledgments

This work was supported by grants R01 EY019474, and P30 EY10572 from the National Institutes of Health, by unrestricted departmental funding from Research to Prevent Blindness, and by the Malcolm Marquis Innovation Fund.

References

[1] Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. *arXiv preprint arXiv:2005.00928*, 2020. 2

[2] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. *Advances in neural information processing systems*, 31, 2018. 2, 5, 7, 8

[3] Esra Ataer-Cansizoglu, Veronica Bolon-Canedo, J Peter Campbell, Alican Bozkurt, Deniz Erdogmus, Jayashree Kalpathy-Cramer, Samir Patel, Karyn Jonas, RV Paul Chan, Susan Ostmo, et al. Computer-based image analysis for plus disease diagnosis in retinopathy of prematurity: performance of the "i-rop" system and image features associated with expert diagnosis. *Translational vision science & technology*, 4 (6):5–5, 2015. 1, 6, 12

[4] Soufiane Belharbi, Aydin Sarraf, Marco Pedersoli, Ismail Ben Ayed, Luke McCaffrey, and Eric Granger. F-cam: Full resolution class activation maps via guided parametric upscaling. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3490–3499, 2022. 2

[5] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3 (1):1–122, 2011. 2, 3, 4

[6] James M Brown, J Peter Campbell, Andrew Beers, Ken Chang, Susan Ostmo, RV Paul Chan, Jennifer Dy, Deniz Erdogmus, Stratis Ioannidis, Jayashree Kalpathy-Cramer, et al. Automated diagnosis of plus disease in retinopathy of prematurity using deep convolutional neural networks. *JAMA ophthalmology*, 136(7):803–810, 2018. 1, 4

[7] Antonin Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical imaging and vision*, 20:89–97, 2004. 3

[8] Hila Chefer, Shir Gur, and Lior Wolf. Generic attention-model explainability for interpreting bi-modal and encoder-decoder transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 397–406, 2021. 2

[9] Hila Chefer, Shir Gur, and Lior Wolf. Transformer interpretability beyond attention visualization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 782–791, 2021. 2

[10] Jianbo Chen, Le Song, Martin Wainwright, and Michael Jordan. Learning to explain: An information-theoretic perspective on model interpretation. In *International conference on machine learning*, pages 883–892. PMLR, 2018. 3

[11] Michael F Chiang, Graham E Quinn, Alistair R Fielder, Susan R Ostmo, RV Paul Chan, Audina Berrocal, Gil Binenbaum, Michael Blair, J Peter Campbell, Antonio Capone Jr, et al. International classification of retinopathy of prematurity. *Ophthalmology*, 128(10):e51–e68, 2021. 1, 4, 12

[12] John Duchi, Shai Shalev-Shwartz, Yoram Singer, and Tushar Chandra. Efficient projections onto the l 1-ball for learning in high dimensions. In *Proceedings of the 25th international conference on Machine learning*, pages 272–279, 2008. 4

[13] Ruth Fong, Mandela Patrick, and Andrea Vedaldi. Understanding deep networks via extremal perturbations and smooth masks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2950–2958, 2019. 2, 3, 5, 6, 7, 8, 14, 15, 17, 19, 20

[14] Ruth C Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *Proceedings of the IEEE international conference on computer vision*, pages 3429–3437, 2017. 1, 2, 3

[15] Clare Gilbert and Allen Foster. Childhood blindness in the context of vision 2020: the right to sight. *Bulletin of the World Health Organization*, 79(3):227–232, 2001. 1

[16] Aric Hagberg, Pieter Swart, and Daniel S Chult. Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008. 5, 12

[17] Adam M Hanif, Sara Beqiri, Pearse A Keane, and J Peter Campbell. Applications of interpretability in deep learning models for ophthalmology. *Current opinion in ophthalmology*, 32(5):452, 2021. 1

[18] Marian-Daniel Iordache, José M Bioucas-Dias, and Antonio Plaza. Total variation spatial regularization for sparse hyperspectral unmixing. *IEEE Transactions on Geoscience and Remote Sensing*, 50(11):4484–4502, 2012. 3

[19] Tong Jian, Yifan Gong, Zheng Zhan, Runbin Shi, Nasim Soltani, Zifeng Wang, Jennifer Dy, Kaushik Chowdhury, Yanzhi Wang, and Stratis Ioannidis. Radio frequency fingerprinting on the edge. *IEEE Transactions on Mobile Computing*, 21(11):4078–4093, 2021. 3

[20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 4, 5, 14

[21] Narine Kokhlikyan, Vivek Miglani, Miguel Martin, Edward Wang, Bilal Alsallakh, Jonathan Reynolds, Alexander Melnikov, Natalia Kliushkina, Carlos Araya, Siqi Yan, et al. Captum: A unified and generic model interpretability library for pytorch. *arXiv preprint arXiv:2009.07896*, 2020. 5, 14

[22] Yann LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998. 2, 5, 14

[23] Yann LeCun et al. Lenet-5, convolutional neural networks. *URL: http://yann. lecun. com/exdb/lenet*, 20(5):14, 2015. 5, 14

[24] Cong Leng, Zesheng Dou, Hao Li, Shenghuo Zhu, and Rong Jin. Extremely low bit neural network: Squeeze the last bit out with admm. In *Proceedings of the AAAI conference on artificial intelligence*, 2018. 3

[25] Dimitrios Lenis, David Major, Maria Wimmer, Astrid Berg, Gert Sluiter, and Katja Bühler. Domain aware medical im-

age classifier interpretation by counterfactual impact analysis. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2020: 23rd International Conference, Lima, Peru, October 4–8, 2020, Proceedings, Part I 23*, pages 315–325. Springer, 2020. 3

[26] Hongjia Li, Ning Liu, Xiaolong Ma, Sheng Lin, Shaokai Ye, Tianyun Zhang, Xue Lin, Wenyao Xu, and Yanzhi Wang. Admm-based weight pruning for real-time deep learning acceleration on mobile devices. In *Proceedings of the 2019 on Great Lakes Symposium on VLSI*, pages 501–506, 2019. 3

[27] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017. 5, 6, 8, 13, 14, 15, 17, 19, 20

[28] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 5

[29] Vitali Petsiuk, Abir Das, and Kate Saenko. Rise: Randomized input sampling for explanation of black-box models. *arXiv preprint arXiv:1806.07421*, 2018. 5

[30] 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, 2, 5, 6, 8, 13, 14, 15, 17, 19, 20

[31] Michael Ryan, Susan Ostmo, Karyn Jonas, Audina Berrocal, Kimberly Drenser, Jason Horowitz, Thomas Lee, Charles Simmons, María Martínez-Castellanos, Robison Chan, and Michael Chiang. Development and evaluation of reference standards for image-based telemedicine diagnosis and clinical research studies in ophthalmology. *AMIA ... Annual Symposium proceedings / AMIA Symposium. AMIA Symposium*, 2014:1902–10, 2014. 4

[32] 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. 2, 5, 6, 7, 8, 13, 14, 19, 20

[33] Avanti Shrikumar, Peyton Greenside, Anna Shcherbina, and Anshul Kundaje. Not just a black box: Learning important features through propagating activation differences. *arXiv preprint arXiv:1605.01713*, 2016. 2, 5, 6, 7, 8, 14, 19, 20

[34] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *International conference on machine learning*, pages 3145–3153. PMlR, 2017. 13

[35] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013. 1, 2, 5, 6, 7, 8, 13, 14, 19, 20

[36] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pages 3319–3328. PMLR, 2017. 2, 5, 6, 8, 13, 14, 19, 20

[37] Elifnur Sunger, Beyza Kalkanli, Veysi Yildiz, Tales Imbiriba, Peter Campbell, and Deniz Erdogmus. Tubular curvature filter: Implicit pointwise curvature calculation method for tubular objects. *arXiv preprint arXiv:2311.11931*, 2023. 6, 12

[38] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 1, 4, 7

[39] Reza Takapoui, Nicholas Moehle, Stephen Boyd, and Alberto Bemporad. A simple effective heuristic for embedded mixed-integer quadratic programming. *International journal of control*, 93(1):2–12, 2020. 3

[40] Peng Tian, Esra Ataer-Cansizoglu, Jayashree Kalpathy-Cramer, Susan Ostmo, Karyn Jonas, RV Paul Chan, J Peter Campbell, Michael F Chiang, and Deniz Erdogmus. Toward a severity index for rop: An unsupervised approach. In *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 1312–1315. IEEE, 2016. 1

[41] Peng Tian, Yuan Guo, Jayashree Kalpathy-Cramer, Susan Ostmo, John Peter Campbell, Michael F Chiang, Jennifer Dy, Deniz Erdogmus, and Stratis Ioannidis. A severity score for retinopathy of prematurity. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1809–1819, 2019. 6, 12

[42] Hristina Uzunova, Jan Ehrhardt, Timo Kepp, and Heinz Handels. Interpretable explanations of black box classifiers applied on medical images by meaningful perturbations using variational autoencoders. In *Medical Imaging 2019: Image Processing*, page 1094911. International Society for Optics and Photonics, SPIE, 2019. 3

[43] Ji Wang, Jie Ji, Mingzhi Zhang, Jian-Wei Lin, Guihua Zhang, Weifen Gong, Ling-Ping Cen, Yamei Lu, Xuelin Huang, Dingguo Huang, et al. Automated explainable multidimensional deep learning platform of retinal images for retinopathy of prematurity screening. *JAMA network open*, 4(5): e218758–e218758, 2021. 2

[44] Daniel E Worrall, Clare M Wilson, and Gabriel J Brostow. Automated retinopathy of prematurity case detection with convolutional neural networks. In *International Workshop on Deep Learning in Medical Image Analysis*, pages 68–76. Springer, 2016. 1

[45] Qiaowei Wu, Yijun Hu, Zhenyao Mo, Rong Wu, Xiayin Zhang, Yahan Yang, Baoyi Liu, Yu Xiao, Xiaomin Zeng, Zhanjie Lin, et al. Development and validation of a deep learning model to predict the occurrence and severity of retinopathy of prematurity. *JAMA Network Open*, 5(6): e2217447–e2217447, 2022. 2

[46] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017. 2, 5, 14

[47] Kaidi Xu, Sijia Liu, Pu Zhao, Pin-Yu Chen, Huan Zhang, Quanfu Fan, Deniz Erdogmus, Yanzhi Wang, and Xue Lin. Structured adversarial attack: Towards general implementation and better interpretability. *arXiv preprint arXiv:1808.01664*, 2018. 3

[48] Ilkay Yildiz, Jennifer Dy, Deniz Erdogmus, Jayashree Kalpathy-Cramer, Susan Ostmo, J Peter Campbell, Michael F Chiang, and Stratis Ioannidis. Fast and accurate ranking regression. In *International Conference on Artificial Intelligence and Statistics*, pages 77–88. PMLR, 2020. 3

[49] Veysi Yildiz, Jennifer Dy, Peter Campbell, Susan Ostmo, Michael Chiang, Stratis Ioannidis, and Deniz Erdogmus. Structurally guided channel attention networks: Sgca-net. In *The 14th PErvasive Technologies Related to Assistive Environments Conference*, pages 93–96, 2021. 1

[50] Veysi Yildiz, Stratis Ioannidis, Ilkay Yildiz, Peng Tian, John Peter Campbell, Susan Ostmo, Jayashree Kalpathy-Cramer, Michael F Chiang, D Erdoğmuş, and J Dy. Structural visual guidance attention networks in retinopathy of prematurity. In *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pages 353–357. IEEE, 2021. 1

[51] Jinsung Yoon, James Jordon, and Mihaela van der Schaar. Invase: Instance-wise variable selection using neural networks. In *International Conference on Learning Representations*, 2018. 3

[52] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13*, pages 818–833. Springer, 2014. 2, 5, 6, 7, 8, 14, 15, 17, 19, 20

[53] Tianyun Zhang, Shaokai Ye, Kaiqi Zhang, Jian Tang, Wujie Wen, Makan Fardad, and Yanzhi Wang. A systematic dnn weight pruning framework using alternating direction method of multipliers. In *Proceedings of the European conference on computer vision (ECCV)*, pages 184–199, 2018. 3, 4