# Fast Hierarchical Games for Image Explanations

Jacopo Teneggi, Alexandre Luster, and Jeremias Sulam, *Member, IEEE*

**Abstract**—As modern complex neural networks keep breaking records and solving harder problems, their predictions also become less and less intelligible. The current lack of interpretability often undermines the deployment of accurate machine learning tools in sensitive settings. In this work, we present a model-agnostic explanation method for image classification based on a hierarchical extension of Shapley coefficients–*Hierarchical Shap (h-Shap)*–that resolves some of the limitations of current approaches. Unlike other Shapley-based explanation methods, h-Shap is scalable and can be computed without the need of approximation. Under certain distributional assumptions, such as those common in multiple instance learning, h-Shap retrieves the exact Shapley coefficients with an exponential improvement in computational complexity. We compare our hierarchical approach with popular Shapley-based and non-Shapley-based methods on a synthetic dataset, a medical imaging scenario, and a general computer vision problem, showing that h-Shap outperforms the state of the art in both accuracy and runtime. Code and experiments are made publicly available.

**Index Terms**—Interpretable Machine Learning, Shapley coefficients, Image Explanations.

## 1 Introduction

Explainability has become a question of increasing relevance in machine learning, where the growing complexity of deep neural networks often renders them *opaque* to us, the humans interacting with them. This issue is commonly referred to as the *black-box problem* and comprises theoretical, technical, and regulatory questions [1], [2]. As deep neural networks take on sensitive tasks in medical, legal, and financial settings, they need to achieve both high accuracy and high transparency for a safe deployment. For example, uninterpretable predictions could mislead clinicians in their decision making rather than support it [3]. Furthermore, it is sometimes required by law [4] to provide an explanation of how data lead an automated algorithm, for example, to reject a loan application [4], [5], [6]. Finally, opaque models can conceal dataset bias, and lead to socially unfair models [7].

In this work, we are particularly interested in explaining models in supervised learning scenarios in order to gain further insights about the concept related to a specific response. For example, assume one has a model that predicts the presence of brain tumor in MRI scans with very high accuracy. What are the most relevant morphological features that indicate the presence of tumor, and where are they located? Can we discover new features of the disease from what the model has learned? Many important problems of this kind exist, but the necessary tools to answer these questions effectively and efficiently are still lacking.

The foundational work by Ribeiro et al. [8] spurred exciting advances in local feature attribution methods, such as Grad-CAM [9], Integrated Gradients [10], and DeepLIFT [11]. Lundberg and Lee [12] provide a unified framework for several different approaches under their SHAP method, which leverages Shapley coefficients–a game-theoretic measure [13]–and feature removal strategies. Unlike other perturbation-based alternatives [14], these methods enjoy of important consistency results and theoretical properties that the resulting attributions satisfy. Since then, a plethora of different explanation methods has been developed[1] for different kinds of data (tabular, sequential, imaging), both based on Shapley coefficients [16] as well as other information theoretic quantities [17], [18], [19]. Although previous work explores structured and hierarchical approaches [16], [20], [21], they remain limited for high-dimensional data.

Notwithstanding the recent advances in image attribution methods based on Shapley coefficients, several limitations hinder their use for "large" images–a standard image contains $\approx 10^6$ pixels, and larger images are used in several important applications. We focus on problems that satisfy a certain *multiple instance learning* assumption [22], which can be found in many relevant fields. We show that in these problems, the computation of Shapley coefficients can be solved efficiently and without the need of approximation by exploring a hierarchical partition of the input image. The contribution of this work is three-fold: first, we present a fast explanation method based on Shapley coefficients that is exponentially faster than popular SHAP methods. Second, under some distributional assumptions similar to those in multiple instance learning problems, we show that the coefficients provided by h-Shap are exact, and can be further approximated in a controlled manner by trading off computational cost. Third, we compare h-Shap with other popular explanation methods on three benchmarks, of varied complexity and dimension, demonstrating that h-Shap

- *J. Teneggi and J. Sulam are with the Mathematical Institute for Data Science and the Department of Biomedical Engineering at the Johns Hopkins University, Baltimore, MD, 21218.*
  *E-mail: jtenegg1@jhu.edu , jsulam1@jhu.edu*
- *A. Luster is with the School of Life Sciences at the Ecole Polytechnique Fédérale de Lausanne, CH-1015, Lausanne.*

1. To our knowledge, Covert et al. [15] compiled the most comprehensive review of currently available explanation methods based on feature removal.

outperforms the state of the art both in terms of runtime and retrieval of relevant features in all experiments.

This paper is organized as follows. In Sec. 2 we briefly summarize the necessary background. We present h-Shap in Sec. 3, including results on computational complexity and approximation. We present experiments in Sec. 4 and their results in Sec. 5. Finally, we discuss our limitations in Sec. 6, and we conclude in Sec. 7.

## 2 Background

In supervised learning scenarios, we are interested in approximating a response or label, $Y \in \mathcal{Y}$, from a given input random sample $X \in \mathcal{X}$. Herein we assume a realizable setting where the response $Y = f^*(X) \in \mathcal{Y}$, for some $f^* : \mathcal{X} \to \mathcal{Y}$, and denote the joint distribution of $(X, Y)$ as $\mathcal{D}$. We look for a function $f : \mathcal{X} \to \mathcal{Y}'$ that approximates $f^*(X)$. Given a loss function $L : \mathcal{Y} \times \mathcal{Y}' \to \mathbb{R}$ that penalizes the dissimilarity between the predicted and real label, we look for $f$ in a suitable functional class with minimal risk, $\mathcal{R} = \mathbb{E}_{\mathcal{D}}[L(Y, f(X))]$. However, $\mathcal{D}$ is typically unknown and instead we are provided with a training set $\{(X^{(i)}, Y^{(i)})\}_{i=1}^{N}$ of observed data. As a result, we search for a function that minimizes the empirical risk,

$$\hat{f} = \arg\min_{f \in \mathcal{F}} \frac{1}{N} \sum_{i=i}^{N} L(Y^{(i)}, f(X^{(i)})), \tag{1}$$

where $\mathcal{F} = \{f_\theta : \theta \in \Theta\}$, with parameters $\theta$ (such as a neural network model). We focus on binary classification problems, where $\mathcal{Y} = \{0, 1\}$ and $\mathcal{Y}' \in \mathbb{R}$, though our general methodology is applicable to multi-class settings as well. We will refer to images–matrices of size $(\sqrt{n} \times \sqrt{n})$–as vectors in the $n$-dimensional real space, i.e. $\mathcal{X} \subseteq \mathbb{R}^n$.

### 2.1 Explaining predictions via Shapley coefficients

Modern machine learning models, in particular those based on deep neural networks, can often provide solutions that perform remarkably well. In many settings, however, one would like to know the contribution of $x_i$, the $i^{th}$ entry of $X$, towards the output. Let us define by $C$ a subset of the entries of $X$, so that $C \subseteq [n] \coloneqq \{1, \dots, n\}$, and define $X_C \in \mathbb{R}^n$ the input that coincides with $X$ in the entries denoted by $C$ but takes a different, *baseline*, value in its complement, $\bar{C}$. In the context of interpretability, we look for a vector $\Phi_{(X,\hat{f})} \in \mathbb{R}^n$, where the $i^{th}$ coordinate reflects the importance of $x_i$ in producing the output $\hat{f}(X)$. Broadly speaking, the features in $C$ provide an explanation for the local prediction $\hat{f}(X)$ if $\hat{f}(X) \approx \hat{f}(X_C)$. Different measures of importance have been proposed to study model interpretability, and thus to compute $\Phi_{(X,\hat{f})}$. In this work we focus on the general approach presented originally by [12] that employs Shapley coefficients [13] as the measure of contribution of every pixel toward the output, which has gained great popularity [23]. We now briefly introduce some game theory notation to define Shapley coefficients.

Let $g = (X, f, [n])$ be an $n$-person cooperative game with players $[n]$ and characteristic function $f : \mathcal{X} \mapsto \mathbb{R}$ which maps the input space $\mathcal{X}$ to a score. In particular, $f(X_C)$ is the score that the players in $C$ would earn by collaborating in the game, with $f(X_\emptyset) = 0$ by convention[2]. A *solution concept* is a rule that assigns a fair contribution to each player in the game. Notably, Shapley coefficients, denoted by $\phi_1(f), \dots, \phi_n(f)$, are the only solution concept of $(X, f, [n])$ that simultaneously satisfy the properties of efficiency, linearity, symmetry, and nullity [13]. In the context of model explanations, input features are regarded as players, and these properties imply that: **i**) feature attributions sum up to the model prediction; **ii**) the attributions of features playing a convex combination of games are equal to the convex combination of the attributions of the features playing the individual games independently; **iii**) the attributions of irrelevant features are simply 0; and **iv**) the attributions of equally important features are equal, respectively. These equip Shapley-based methods with a useful set of properties, which are not generally satisfied by others attributions methods [14].

Shapley coefficients can be derived axiomatically [13], and they are defined as

$$\phi_i(f) = \sum_{C \subseteq [n] \setminus \{i\}} \frac{|C|!(n - |C| - 1)!}{n!} \left[ f(X_{C \cup \{i\}}) - f(X_C) \right]. \tag{2}$$

This way, $\phi_i(f)$ represents the averaged marginalized contribution of $x_i$ over all possible subsets of $[n]$. Eq. (2) also illustrates what is arguably the most important limitation of Shapley coefficients: their computational cost is exponential in the dimension of the input features (as there are exponentially many distinct subsets $C$) requiring $2^n$ unique evaluations of $f$. This quickly becomes intractable in image classification problems when $f$ is a convolutional neural network and $n \approx 10^6$, or larger. As a result, all state-of-the-art image explanation methods based on Shapley coefficients rely on some approximation strategy to work around this computational limitation. For instance, GradientExplainer [12] extends Integrated Gradients [10] by sampling multiple references from the background dataset to integrate on. Similarly, DeepExplainer [12], [25] builds upon DeepLIFT [11] by choosing a per-node attribution rule that can approximate Shapley coefficients when integrated over many background samples. Finally, PartitionExplainer employs a hierarchical clustering approach inspired by Owen coefficients [26], [27], [28], which generalize Shapley coefficients to cooperative games with *a-priori* coalition structures. Given a game $(X, f, [n])$, let $\mathcal{G} = \{G_1, \dots, G_m\}$ be a coalition structure such that $\bigcup_{G \in \mathcal{G}} = [n]$, and $G_q \cap G_u = \emptyset$ for $q \neq u$. Then, the Owen coefficient of $x_i$ is defined as

$$\varphi_i(f) = \sum_{\substack{H \subseteq [m] \setminus \{q^*\} \\ C \subseteq G_{q^*} \setminus \{i\}}} w_{H,M,C,G_{q^*}} \left[ f(X_{Q \cup C \cup \{i\}}) - f(X_{Q \cup C}) \right], \tag{3}$$

where $w_{H,M,C,G_{q^*}}$ is an appropriate constant, $[m] \coloneqq \{1, \dots, m\}$, $Q = \bigcup_{q \in H} G_q$, and $i \in G$, $q^* \in [m]$. Similarly to Shapley coefficients, Owen coefficients are the only solution concept that satisfy similar properties of efficiency, marginality, and symmetry both across and within coalitions [29]. Intuitively, when looking at feature $i$ from the perspective

2. Formally speaking, game theory [24] requires a characteristic function $v : \mathcal{P}(X) \to \mathbb{R}$, where $\mathcal{P}(X)$ is the power set of $X$. Herein, and following prior work [12], we assume $v(C) = f(X_C)$, $\forall C \subseteq X$, and therefore use $f$ for the sake of simplicity.

of Shapley coefficients (i.e. Eq. (2)), one has to consider all possible subsets of the remaining players. On the other hand, when considering the Owen coefficient of feature $i$ in coalition $G_{q^*}$ (i.e. Eq. (3)), one can only observe other coalitions participate in the game together as a whole, while still being able to observe all possible subsets of players within coalition $G_{q^*}$. This *a-priori* coalition structure reduces the number of subsets of players to explore. Given the close relation between PartitionExplainer and h-Shap, we include a detailed comparison in Appendix C.

To conclude, while the methods above provide approximations that can sometimes work in practice, they only provide consistency results and lack accuracy guarantees when they are run with a few model evaluations [19]. Hence, it is hard to understand when they will and will not be effective. We will compare extensively with these approaches later in Sec. 4.

We remark that one of the most important details of any explanation method based on feature removal is the baseline, which defines the value that $X_C$ takes in the entries not in $C$. There are different approaches to removing features, ranging from using the default value of 0, to using their conditional distribution (refer to [15] for further details). Computing the latter can be challenging, and recent work has explored various approximations [30], [31]. The effects of using different baselines have also been investigated in images [32] and tabular data [33]. We follow the standard approach of setting the baseline to the unconditional expected value over the training dataset [12], [34], and comment on potential extensions later.

### 2.2 Multiple Instance Learning

In this work, we focus on problems with particular joint distributions of samples and labels. Our guarantees will apply to settings broadly known as *Multiple Instance Learning* (MIL) [22], [35]. In MIL, each *instance* $x_i$ is assumed to have an instance-label, and the sample $X$ is regarded as a *bag* that aggregates all instances. The bag, $X$, has its own label $Y \in \{0, 1\}$ determined by its constituent instances. In its simplest version, the bag is assumed to be positive if at least one of its instances is positive. As an example, an image of cells will be labeled with `infection` if at least one cell in it is `infected`. Importantly, the learner does not have access to the instance-labels, but only to the global label $Y$. Such an MIL setting appears in several important problems [36], [37], [38]. In the context of our work, we assume that the prediction rule satisfies such an MIL assumption. More precisely, we will assume that

$$f^*(X) = 1 \iff \exists\, C \subseteq [n] :\ f^*(X_C) = 1. \tag{4}$$

In words, Eq. (4) implies that $f^*(X)$ will be 1 as soon as there is at least one subset $C$ of $[n]$ that contains the *concept* we are interested in detecting. This is simply a formalization of the setting we were describing earlier, where the concept can be a specific morphological feature in a brain scan, a sick cell in a blood smear, or something as general as a traffic light in a street image.

As a partial summary of this section, $\hat{f}$ is trained to detect a binary concept in a sample image, and we would like to detect which subsets of the input, $X_C$, are relevant for this task. While this could in principle be done via Shapley coefficients, this is computationally intractable. We now move on to present our approach, which will address this limitation.

## 3 Hierarchical-Shap

Our motivating observation is that if an area of an image is uninformative (i.e. it does not contain the concept), so will be its constituent sub-areas. Therefore, the exploration of relevant areas of an image can be done in a hierarchical manner. There exists extensive literature on hierarchies of games and their properties [39], [40]. Our contribution is to deploy these ideas for the purpose of image explanations.

We now make this more precise. Let $\mathcal{T}_0 = (S_0, \mathcal{T}_1, \ldots, \mathcal{T}_\gamma)$ be a recursive $\gamma$-partition tree of $X$, where $S_0$ is the root node containing all features of $X$, i.e. $S_0 = [n], |S_0| = n$, and $\mathcal{T}_1, \ldots, \mathcal{T}_\gamma$ are the subtrees branching off of $S_0$. Let $c(S_i) = \{C_1, \ldots, C_\gamma\}$ denote the children of $S_i$, and $h_{\hat{f}} : S_i \mapsto (X, \hat{f}, c(S_i))$ be a mapping from the node $S_i$ of $\mathcal{T}_i$ to the $\gamma$-person cooperative game $(X, \hat{f}, c(S_i))$. Succinctly, $\mathcal{G}_0 = h_{\hat{f}}(\mathcal{T}_0)$ is a hierarchy of $\gamma$-person games, and we denote by $\phi_{i,1}(\hat{f}), \ldots, \phi_{i,\gamma}(\hat{f})$ the Shapley coefficients of $g_i \in \mathcal{G}_0$. In simpler words, we partition an image $X$ into *a few disjoint components*, compute the Shapley coefficients $\phi_i$ of each component, and then partition further in a hierarchical manner. In particular, the number of such partitions per level (specified by $\gamma$) is very small: if $X$ is a one dimensional vector, we set $\gamma = 2$ and $\mathcal{T}_0$ is a binary tree; when $X$ is a $(\sqrt{n} \times \sqrt{n})$ image, $\gamma = 4$ and $\mathcal{T}_0$ is a quadtree. As a result, computing all $2^\gamma$ unique evaluations of $\hat{f}$ required for each game $(X, \hat{f}, c(S_i))$ is trivial. For images, each coefficient requires only 16 model evaluations. In fact, the remaining coefficients (for the same node) involve the same terms but in different permutations, so no extra model evaluations are needed. We have chosen to employ symmetric disjoint partitions in this work (i.e. halves for vectors, quadrants for images, etc) for simplicity only. More sophisticated (and potentially data-dependent) hierarchical partitions are possible as well. We will comment on this in the discussion.

Given such nested partitions, h-Shap relies on evaluating the resulting hierarchy of games while only visiting nodes that are relevant. More precisely, beginning at $S_0$, it computes the coefficients $\phi_{0,1}, \ldots, \phi_{0,\gamma}$ of $g_0$. Under Eq. (4), if any $\phi_{0,i} = 0$, all features in the corresponding subtrees will also be irrelevant. As a result, they can be ignored altogether, and we only proceed by exploring the $S_i$ for which $\phi_i > 0$. This process finishes when all relevant leaves have been visited. In practice, we introduce two parameters to add flexibility. We set a relevance tolerance, $\tau$, which determines the threshold to be used to declare a partition relevant, and therefore expand on its subtrees. We further introduce a minimal feature size, $s$, that serves as a condition for termination. These two parameters are naturally motivated by application and easy to set. For example, it might not be that useful for a domain expert to know the exact pixel-level explanation of a given input. Rather, it would be more informative to have a coarser aggregation of the features that inform the model prediction. Later in this section, we will precisely characterize how the minimal feature size $s$ affects the

**Algorithm 1** Depth-first h-Shap ($d$h-Shap)

**procedure** $d$H-SHAP($X, \mathcal{T}_0, \hat{f}$)
**inputs:** image $X$, threshold $\tau \geq 0$, trained model $\hat{f}$
  $g_0 \leftarrow (X, \hat{f}, c(S_0))$
  $\phi_{0,1}, \ldots, \phi_{0,\gamma} \leftarrow \texttt{shap}(g_0)$
  **for all** $\phi_i$ **do**
    **if** $\phi_i > \tau$ **then**
      **if** $|S_i| \leq s$ **then**
        **return** $S_i$
      **else**
        **return** $d\texttt{h-Shap}(X, \mathcal{T}_i, \hat{f})$
      **end if**
    **end if**
  **end for**
**end procedure**
$L \leftarrow d\texttt{h-Shap}(X, \mathcal{T}_0, \hat{f})$

dissimilarity between h-Shap's attributions and the exact Shapley coefficients. On the other hand, model deviations and noise in the input may result in positive coefficients very close to 0. Requiring $\phi_i > \tau > 0$ provides control over the sensitivity of the method. Finally, when $\tau = 0, s = 1$, h-Shap simply explores all relevant nodes in $\mathcal{T}_0$ as described above.

Fixed $\tau$ and $s$, h-Shap explores $\mathcal{T}_0$ starting from $S_0$, and it visits all relevant nodes $S_i : \phi_i > \tau, |S_i| \geq s$. This tree exploration can be naturally done in a depth-first or breadth-first manner; Algorithm 1 presents $d$h-Shap (depth-first h-Shap). Please refer to Algorithm 2 in Appendix A for $b$h-Shap (breadth-first h-Shap). The only difference between the two algorithms is that the former defines $\tau$ as an absolute value (e.g. 0), whereas the latter does so relative to the pooled Shapley coefficients of all nodes at the same depth (e.g. $50^{th}$ percentile). Both algorithms return the set of relevant leaves $L \subseteq [n]$ with coefficients greater than $\tau$, and the saliency map $\widehat{\Phi}_{(X,\hat{f})}$ is finally computed as

$$\widehat{\phi}_i = \begin{cases} 1/|L| & \text{if } i \in L, \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

This choice will ensure that $\widehat{\Phi}_{(X,\hat{f})}$ is consistent with the exact Shapley attributions $\Phi_{(X,\hat{f})}$ under the MIL assumption, as we will formalize shortly.

To mask features out (i.e. as baseline), h-Shap uses their expected value (or *unconditional distribution* [34]) for simplicity, as done by other works [15]. As pointed out by [12], [15], this is valid under the assumptions of model linearity and feature independence[3]. Yet, as we will argue later in Sec. 7, the feature independence property holds approximately in the cases we are interested in this work, whereas our MIL assumptions are enough to provide specific guarantees without requiring linearity of the model. We will also show in Sec. 4 that these assumptions are sufficient for h-Shap to work well in practice. More generally, our contribution is independent of the particular method employed for sampling the baseline, and follow-up work can employ better approximations of both the observational and interventional conditional distributions in appropriate tasks [41].

3. We refer to [19], [23], [34], [41] for recent discussion on the use of *observational* vs *interventional* conditional distributions in the context of removal-based explanation methods.

### 3.1 Computational analysis

The benefit of h-Shap relies in decoupling the dimensionality of the sample $X$ (i.e. $n$), from the number of players in each game (i.e. $\gamma$). As we will explain in this section, this leads to an exponential computational advantage over the general expression in Eq. (2) in explaining $\hat{f}$. In the analysis that follows, we do not include the computation of the baseline value–which we assume fixed, see discussion in Sect. 7–and we refer the reader to the proofs of all the results in this section to the Appendix B. Let us denote by $\hat{\mathcal{T}}_0$ the subtree of $\mathcal{T}_0$ explored by h-Shap (i.e. the one with the visited nodes only). We will also assume in this section that $n$ is a power of $\gamma$ for simplicity of the expressions[4]. We begin by making the following remark.

**Remark 3.1** (Computational cost)**.** *Given $X \in \mathbb{R}^n$, h-Shap requires at most $2^\gamma k \log_\gamma(n)$ model evaluations, where $k$ is the number of relevant leaves in $\hat{\mathcal{T}}_0$.*

This result follows directly by noting that the cost of splitting each node is always $2^\gamma$, and by realizing that each important leaf takes, at most, $\log_\gamma(n)$ nodes, which is exponentially better than the cost of Eq. (2). The reader should recall that the number of internal nodes of a full and complete $\gamma$-partition tree is $(n-1)/(\gamma-1)$. Then, the above result is relevant whenever $k \log_\gamma n < (n-1)/(\gamma-1)$. This implies that further benefit is obtained whenever $k = \mathcal{O}(n/\log_\gamma n)$, which is only a mild requirement in the number of relevant features.

Moreover, it is of interest to know the expected computational cost, which can be significantly smaller than the upper bound above. Throughout the rest of this section, and to provide more precise results, we will let the data $X$ be drawn from a distribution of *important* and *non-important* features. A distribution is "important" in the sense that it leads to positive responses.

**Assumption A1.** *The data $X \in \mathbb{R}^n$ is drawn so that each entry $x_i \sim a_i\mathcal{I} + (1-a_i)\mathcal{I}^c$, where $a_i \sim$ Bernoulli$(\rho)$ is a binary random variable that indicates whether the feature $x_i$ comes from an* important *distribution $\mathcal{I}$, or its* non-important *complement $\mathcal{I}^c$, so that*

$$\hat{f}(X_C) = 1 \iff \exists i \in C : \; x_i \sim \mathcal{I}, \; C \subseteq [n]. \tag{6}$$

With these elements, we present the following result.

**Theorem 3.2** (Expected number of visited nodes)**.** *Assume $X$ and $\hat{f}(X)$ satisfy A1, $\tau = 0$, and $s = 1$. Then, the expected number of visited nodes in $\hat{\mathcal{T}}_0$ is*

$$\mathbb{E}[|\hat{\mathcal{T}}_0|] = 1 + \gamma(1 - p(S_0))\mathbb{E}[|\hat{\mathcal{T}}_1|], \tag{7}$$

*where*

$$p(S_i) = \begin{cases} (1-\rho)^{\frac{|S_i|}{\gamma}} & \text{if } i = 0, \\ (1-\rho)^{\frac{|S_i|}{\gamma}} \left( \frac{1-(1-\rho)^{|S_i|\frac{\gamma-1}{\gamma}}}{1-(1-\rho)^{|S_i|}} \right) & \text{otherwise.} \end{cases}$$

4. Note that it is trivial to accommodate cases where this is not true.

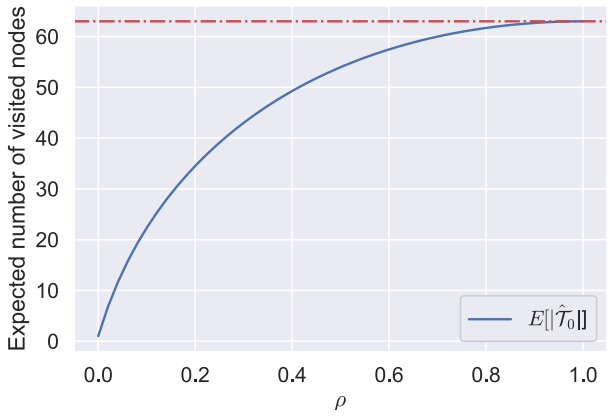



Fig. 1: Expected number of visited nodes as a function of $\rho$ when $n = 64, \gamma = 2, s = 1$.

See Proof B.1. This result does not provide a closed-form expression for the expected number of visited nodes (and, correspondingly, computational cost), but it does provide a simple recurrent formula that can be easily computed. Naturally, this cost depends on the Bernoulli probability $\rho$, the average number of important features in $X$. We present the resulting $\mathbb{E}[|\hat{\mathcal{T}}_0|]$ for a specific case in Fig. 1 as a function of $\rho$, showing that indeed the expected cost can be much lower than the worst-case bound. While this result (and, centrally, Assumption A1) was presented for the case where the relevant features are of size 1, similar results can be provided for the case when the minimal features size $s > 1$.

### 3.2 Accuracy and Approximation

Recall that h-Shap provides image attributions by means of a hierarchy of collaborative games. As a result, the attributions are different, in general, from those estimated by analyzing the grand coalition directly–that is, by the general Shapley approach in Eq. (2). We remark that computing the Shapley coefficients directly from Eq. (2) quickly becomes intractable in image classification tasks. For example, even for a toy-like dataset of small $10 \times 10$ pixels images, assuming that each model computation takes 1 nanosecond (which is unrealistically fast), computing the exact Shapley coefficients would take $\approx 3 \times 10^{13}$ years. Yet, we now show that under A1, h-Shap can in fact provide exact Shapley coefficients while being exponentially faster. Additionally, h-Shap can provide controlled approximations by trading computational efficiency with accuracy.

We begin by noting that under the MIL assumption, all positive features have the same importance. This agrees with intuition that the number of times the positive concept appears in the input image does not affect its label. We denote as $\Phi$ and $\widehat{\Phi}$ the exact and hierarchical Shapley coefficients, respectively, for simplicity.

**Remark 3.3.** *Under A1, and denoting $k = \|\Phi\|_0$, it holds that the exact saliency map $\Phi$ satisfies*

$$\phi_i = \begin{cases} 1/k & \text{if } x_i \sim \mathcal{I} \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

This remark follows simply from the nullity and symmetry properties of Shapley coefficients. As a result, the saliency map computed by h-Shap, $\widehat{\Phi}$, as in Eq. (5), coincides with $\Phi$ under the MIL assumption. We now derive a more general similarity lower bound between $\Phi$ and $\widehat{\Phi}$ that allows for minimal feature sizes $s > 1$. For simplicity, we assume that $n$ and $s$ are powers of $\gamma$, and $1 \leq s \leq n$. First of all, because of the MIL assumption, h-Shap will *only* keep exploring nodes that have at least one important feature in them at each level of the hierarchy. Thus, for each important feature $i$ with $\Phi_i = 1/k$ there will be a non-zero coefficient produced by h-Shap. The following result precisely quantifies to what extent these two vectors $\Phi$ and $\widehat{\Phi}$ match.

**Theorem 3.4** (Similarity lower bound)**.** *Assume $X \in \mathbb{R}^n$ and $\hat{f}(X)$ satisfy A1, and $k = \|\Phi\|_0$. Then*

$$\frac{\langle \Phi, \widehat{\Phi} \rangle}{\|\Phi\|_2 \|\widehat{\Phi}\|_2} \geq \max\{1/\sqrt{s}, \sqrt{k/n}\}. \tag{9}$$

See Proof B.2. This result shows that not only does h-Shap provide faster image attributions, but it retrieves the exact Shapley coefficients defined in Eq. (8) under the MIL assumption if $s = 1$. Notwithstanding, one can employ a larger minimal feature size, $s > 1$, while still providing attributions that are similar to the original ones. In light of the result in Theorem 3.2, the latter attributions will naturally result in improved (smaller) computational costs.

## 4 Experiments

We now move to demonstrate the performance of h-Shap and of other state-of-the-art methods for image attributions. Our objective is mainly to compare with other Shapley-based methods, such as GradientExplainer [12], DeepExplainer [12], [25], and PartitionExplainer[5]. We also include LIME[6] [8] given its relation to Shapley coefficients, and Grad-CAM[7] [9] because of its popularity. We study three complementary binary classification problems of different complexity and input dimension: a simple synthetic benchmark, a medical imaging dataset, and a general computer vision task. We focus on scenarios where the ground truth of the image attributions (i.e. what defines the label) is well defined and available for evaluation. All experiments were conducted on a workstation with NVIDIA Quadro RTX 5000. Our code is made available for the purpose of reproducibility[8]. When possible, each method was set to use as much GPU memory as possible, so as to minimize their runtime. DeepExplainer and GradientExplainer were constrained the most by memory, reflecting their limitation in analyzing large images. We use h-Shap with both an absolute threshold $\tau = 0$, and a relative threshold $\tau$ equal to the $70^{th}$ percentile, which we refer to as $\tau = 70\%$ with abuse of notation. Finally, we perform *full* model randomization sanity checks [42] on the network used in the synthetic dataset for all explanation methods. We refer the reader to Appendix E for these results.

5. The implementation of GradientExplainer, DeepExplainer and PartitionExplainer are openly available at https://github.com/slundberg/shap.
6. https://github.com/marcotcr/lime.
7. https://github.com/jacobgil/pytorch-grad-cam.
8. https://github.com/Sulam-Group/h-shap

This article has been accepted for publication in IEEE Transactions on Pattern Analysis and Machine Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2022.3189849



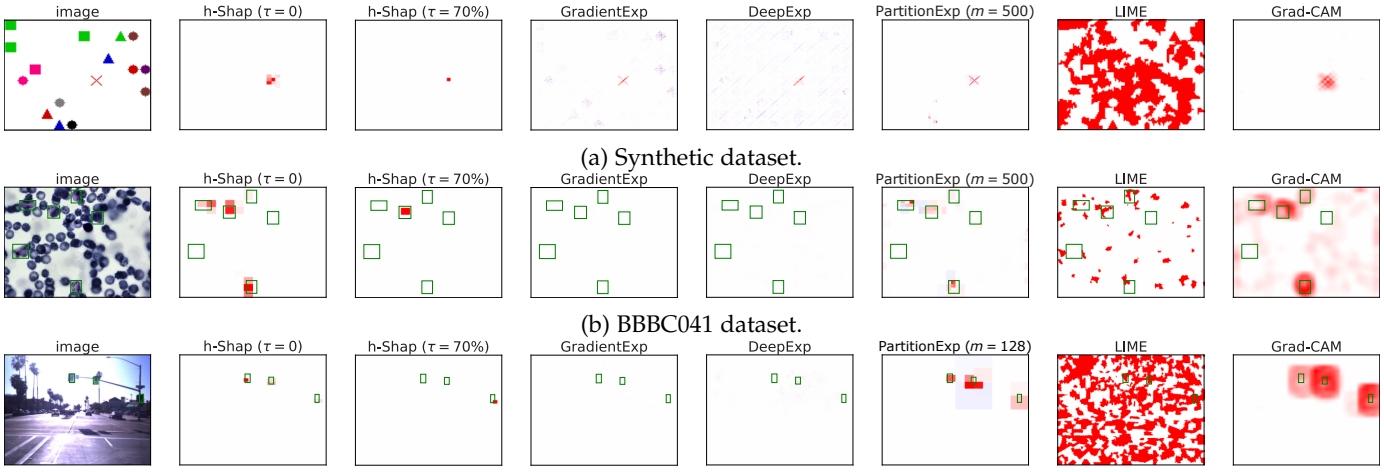



Fig. 2: A few saliency maps for the three settings studied in this work, where blue pixels have negative, white pixels have negligible, and red pixels have positive Shapley coefficients. The color mapping is adapted to each saliency map and centered around 0. For h-Shap, we show the saliency map before the normalization step.

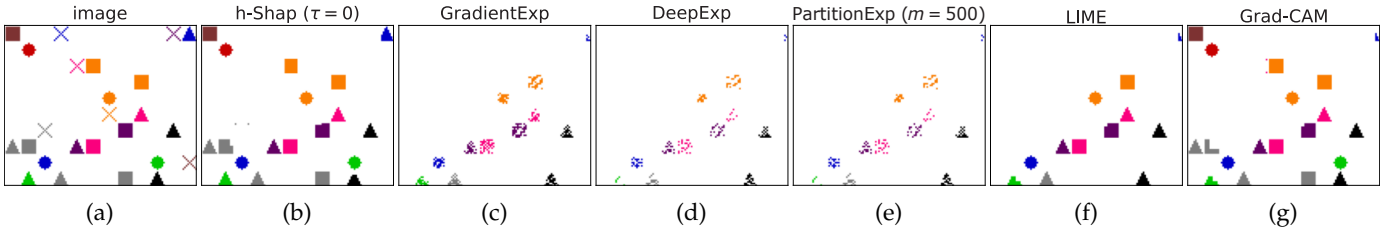



Fig. 3: Ablation examples for all explanation methods removing all important pixels from the original image 3a. The model is trained to predict if a given image does contain a cross or not.

### 4.1 Synthetic dataset

We created a controlled setting where the joint data distribution is completely known, giving us maximal flexibility for sampling. We generate images of size $100 \times 120$ pixels with a random number of non-overlapping geometric shapes of size $10 \times 10$ and of different colors, uniformly distributed across the image. Each image that contains at least one cross receives a positive label, and each image without any crosses receives a negative label. Alongside with the images, we generate the ground truth saliency maps by setting all pixels that precisely lie on a cross to 1, and every other pixel to 0. We generate 8000 positive and negative images, and we randomly sample train, validation, and test splits, with size $5000, 1000$ and $2000$ images, respectively. We train a simple ConvNet architecture, optimizing for 50 epochs with Adam [43], learning rate of $0.001$ and cross-entropy loss. We achieve an accuracy greater than $99\,\%$ on the test set–implying that the model has effectively satisfied the MIL assumption for this problem. From the true positive predictions on the test set, we choose 300 example images with 1 cross and as many with 6 crosses to evaluate the saliency maps. Fig. 2a presents a qualitative demonstration of h-Shap and other related methods on this task.

### 4.2 P. vivax (malaria) dataset

Moving on to a real and high-dimensional problem, we explore the BBBC041v1 dataset, available from the Broad Bioimage Benchmark Collection[9] [44]. The dataset consists of 1328, $1200 \times 1600$ pixels blood smears with uninfected (i.e. red blood cells and leukocytes) and infected (i.e. gametocytes, rings, trophozoites, and schizonts) blood cells. The dataset also comprises bounding-box annotations of both healthy and sick cells. We consider the binary problem of detecting images that contain at least one trophozoite, yielding 655 positive and 673 negative samples. Given the small amount of data available, we augment the training dataset with random horizontal flips, and we randomly choose 120 positive, and equally many negative images as the testing set. We apply transfer learning to a ResNet18 [45] network pretrained on ImageNet. We optimize all parameters of the pretrained network for 25 epochs with Adam [43]–learning rate $0.0001$. We use cross-entropy loss and learning rate decay of $0.2$ every 10 epochs. After training, our model achieves a test accuracy greater than $99\,\%$. We finally aggregate all 112 true positive predictions for evaluation, without distinction on the number of trophozoites in the image. Fig. 2b shows a sample image and the corresponding saliency maps produced by the various methods.

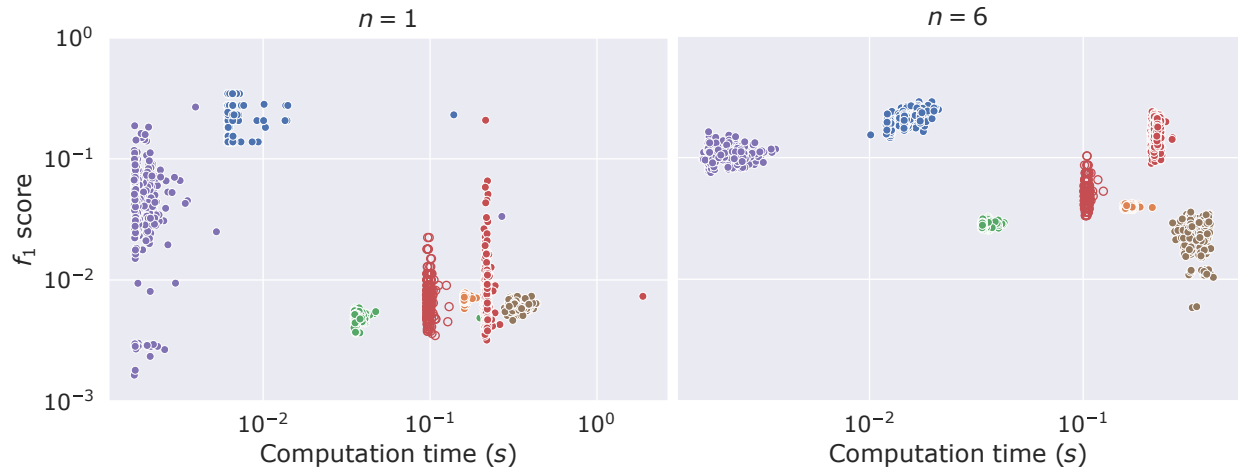
### 4.3 LISA traffic light dataset.

We finally look at a general computer vision dataset consisting of driving sequences collected in San Diego, CA, available

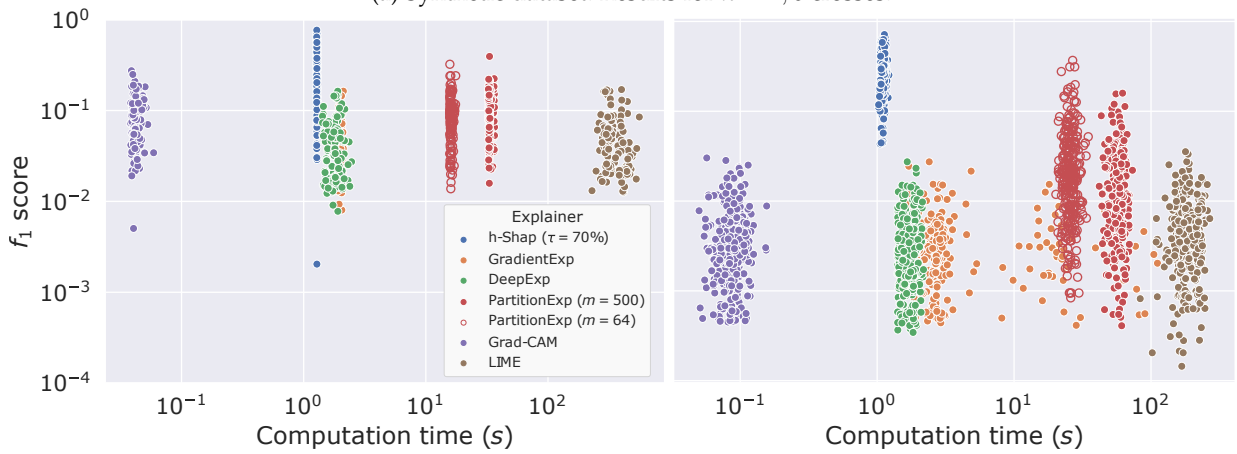
9. https://www.kaggle.com/kmader/malaria-bounding-boxes.

This article has been accepted for publication in IEEE Transactions on Pattern Analysis and Machine Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2022.3189849






(a) Synthetic dataset. Results for $n = 1, 6$ crosses.




(b) BBBC041 dataset.

(c) LISA dataset.

Fig. 4: $f_1$ scores as a function of runtime for all explanation methods in all three experiments. To account for noise in the explanations, we threshold saliency maps at $1 \times 10^{-6}$ and compute $f_1$ scores on the resulting binary masks. For PartitionExplainer, $m$ indicates the maximal number of model evaluations.

from[10] [46], [47]. The complete dataset counts 43 007 frames of size $960 \times 1280$ pixels, and 113 888 annotated traffic lights. From this set, we take daytime traffic images, and train a model to predict the presence of a green light in a sample image. We respect the original train/test splits, providing 6108 train, 3846 test positive samples, and 6667 train, 3627 test negative samples. As before, we use data augmentation and apply transfer learning on a pretrained ResNet18. We optimize all parameters of the pretrained network for 25 epochs with Adam [43]–learning rate 0.0001. We use cross-entropy loss and learning rate decay of 0.2 every 10 epochs. After training, we achieve a test accuracy of $\approx 95\,\%$. Finally, we randomly sample 300 true positive examples to evaluate the different attribution methods on. Fig. 2c illustrates a positive sample image, and the corresponding saliency maps.

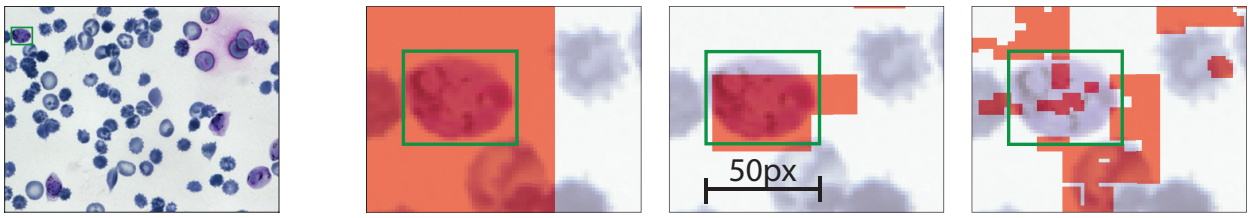
## 5 Results

Fig. 2 shows a visual comparison of some saliency maps obtained in the three experiments (for more examples, see Fig. F.1). Note that while the saliency maps produced by GradientExplainer and DeepExplainer appear empty in Fig. 2b and 2c, they are not, and instead the single pixels are too small to be visible (these are large images). This illustrates how current Shapley-based explanation methods fall short of producing informative saliency maps in problems with large images. We further evaluate the explanation methods by means of three performance measures: ablation tests, accuracy, and runtime.

### 5.1 Ablation tests

As commonly done in literature [12], [32], [33] we remove the top $k$ scoring features of all methods by setting them to their expected value, and plot the logit of the prediction as a function of $k$. For these experiments, we use $\tau = 0$ so as to find *all* the features that are relevant for the model. Fig. 3 shows ablation results on one example image from the synthetic dataset for all explanation methods. We expect a perfect method to remove all crosses from the image–and only those. We can appreciate how h-Shap removes mostly only the crosses, while other methods also erase other shapes which should not be identified as important. Furthermore, removing more relevant features should produce a steeper drop of the prediction logit. We include the respective curves in Fig. F.2, depicting that h-Shap's logit curves either quickly drop towards 0 or provide a logit $\approx 0$ at complete ablation. Indeed, h-Shap quickly identifies the most relevant features

10. https://www.kaggle.com/mbornoe/lisa-traffic-light-dataset.




(a) Original image. (b) $s = 80$ pixels. (c) $s = 20$ pixels. (d) $s = 5$ pixels.

Fig. 5: Degradation of h-Shap's maps as the minimal feature size $s$ becomes smaller than the target concept.

in the image. Naturally, as tasks become harder, the accuracy of $\hat{f}$ decreases, and the model gets further away from the oracle function $f^*$. In these cases (for the real datasets), $\hat{f}$ might not satisfy Eq. (4), resulting in noisier saliency maps, and correspondingly, non-monotonic curves.

### 5.2 Accuracy and Runtime

Since we have ground-truth explanations in all these cases (i.e. a cross, a sick cell, or a green traffic light), we use $f_1$ scores as a measure of goodness of explanation. We argue that $f_1$ scores are a particularly informative measure for explanations (when ground-truth is known), and consistent with previous work [48]. Fig. 4 depicts the $f_1$ scores as a function of runtime for every explanation method and experiment. The advantage of setting a relative relevance tolerance $\tau$ is clear: to detect the *most* relevant features and discard the noisy ones, taking into account the risk of the model $\hat{f}$, while also decreasing runtime. These results reflect how the computational cost and accuracy guarantees described earlier translate into application. Not only does h-Shap decrease runtime compared to current Shapley-based explanation methods–by one to two orders of magnitude–but it also increases the $f_1$ score. Fig. 4a shows that h-Shap's accuracy is not affected by the number of crosses in the image, while other methods deteriorate when there is only one cross to detect in the image. Importantly, in all experiments–both synthetic and real–h-Shap consistently provides more accurate and faster saliency maps compared to other Shapley-based methods, and it is only beaten in speed by Grad-CAM, which provides less accurate saliency maps.

## 6 Discussion

### 6.1 Limitations

Before concluding, we want to delineate the limitations of h-Shap, the most important of which is its MIL assumption on the data distribution. The methodology proposed in this work is designed to identify local *findings* that produce a positive global response, accurately and efficiently. These are precisely the important features $C$ analyzed in Sec. 3. This setting is controlled by the ratio of the size of the actual object that defines the label, and the minimal feature size of the algorithm. As an example, Fig. 5 depicts a zoomed-in version of the map produced by h-Shap for one of the samples from the P. vivax dataset, for different values of $s$. We see that even when $s$ is somewhat smaller than the object, h-Shap still recognizes the important features in the image. Once $s$ is too small, however, the resulting map breaks down, as our assumption does not hold any more. Indeed, small ($5 \times 5$ pixels) image patches break Assumption A1 because a small patch of a cell is not sufficient for the model to recognize it. In practice, these failure cases can easily be identified by deploying simple conditions searching over decreasing sizes of $s$ (which would not increase the computational cost). We note that Eq. (6) can also be phrased as an OR function across features. Intuitively, when the minimal feature size $s$ is smaller that the concept of interest, the OR function is no longer appropriate.

A second limitation of h-Shap pertains the way hierarchical partitions are created. We have chosen to use quadrants for their effectiveness and elegance, but this could be sub-optimal: important features may fall in-between quadrants, impacting performance. This limitation is minor, as it can be easily fixed by applying ideas of cycle spinning and averaging the resulting estimates. Furthermore, and more interestingly, hierarchical data-dependent partitions could also be employed. We regard this as future work.

### 6.2 Baseline and assumptions

Recall that all explanation methods based on feature removal–like Shapley-based explanation methods–are sensitive to the choice of baseline, i.e. the reference value used to mask features. Then, we now turn our attention to h-Shap's masking strategy, or alternatively, how to sample a reference. We recall that in this work we defined the variable $X_C$ as

$$(X_C)_i = \begin{cases} X_i & \text{if } i \in C \\ R_i & \text{otherwise,} \end{cases} \tag{10}$$

where $R \in \mathbb{R}^{n-|C|}$ is a baseline value. Throughout this work, we have treated $R$ as a fixed, deterministic quantity. However, more generally, reference inputs are random variables. Let this masked input be the random variable $X_C = [\bar{X}_C, R] \in \mathbb{R}^n$, where $\bar{X}_C \in \mathbb{R}^{|C|}$ is fixed, and $R$ is a random variable. Here, we want to identify what relationships in the data distribution are important for the model, so we follow the original approach in [12]. Indeed, the definition of Shapley values for the $i^{th}$ coefficient in Eq. (2) can be made more precise by writing its expectation $\mathbb{E}[\hat{f}(X_{C\cup\{i\}}) - \hat{f}(X_C)]$ as

$$\mathbb{E}_R[\hat{f}([\bar{X}_{C\cup\{i\}}, R]) \mid \bar{X}_{C\cup\{i\}}] - \mathbb{E}_R[\hat{f}([\bar{X}_C, R]) \mid \bar{X}_C]. \tag{11}$$

As it can be seen, if the model $\hat{f}$ is linear, and the features are independent, then Eq. (11) simplifies to

$$\hat{f}([\bar{X}_{C\cup\{i\}}, \mathbb{E}[R]]) - \hat{f}([\bar{X}_C, \mathbb{E}[R]]), \tag{12}$$

where $\mathbb{E}[R]$ is an unconditional expectation which can be easily computed over the training data, and is precisely the fixed baseline we employed in this work.

How realistic are these assumptions in our case? First, the cases that we study here approximately satisfy feature

independence in a local sense, and it is therefore reasonable to consider the input features as independent when $s$–the minimal feature size–is greater or similar to the size of the concept we are interested in detecting. Indeed, this is precisely true in the synthetic dataset, where each $10 \times 10$ pixels shape is sampled independently from the others. This assumption is still approximately valid in the other two experiments, where, for example, the presence or absence of a cell does not affect the content of the image so many pixels apart. On the other hand, while we have chosen very general models $\hat{f}$ which are far from linear, we argue that A1 is enough to obtain a weaker sense of interpretability: looking at

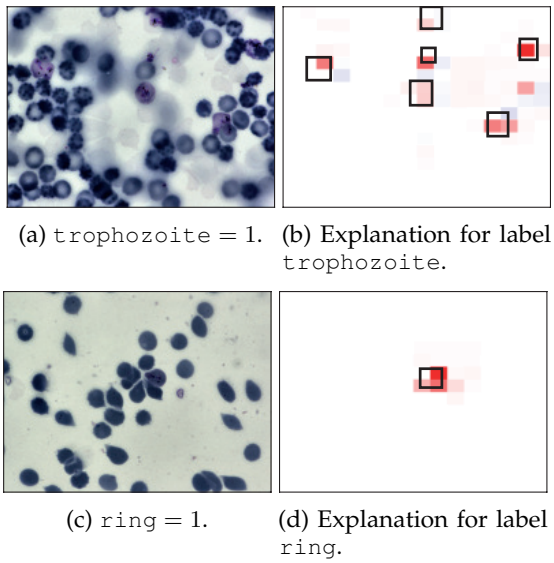
$$\hat{f}([X_C, \mathbb{E}[R]]), \tag{13}$$

and under the MIL assumption, there are only two mutually exclusive events for the subset $C$: (a) $C$ contains at least one relevant feature, and (b) $C$ does not contain any relevant features. When event (a) occurs, Eq. (13) will necessarily yield a high value $\approx 1$, regardless of the value of the baseline $\mathbb{E}[R]$. It follows that if both $C \cup \{i\}$ and $C$ contain important features, Eq. (12) will be $\approx 0$; which agrees with intuition that all important features are equally important. As a result, because $\mathbb{E}[R]$ is fixed and A1 holds, a positive value of Eq.(12) is only attained if (i.e. implies that) $i$ is an important feature (and it also implies that $\mathbb{E}[R]$ is not important).

To summarize, the choice of using the unconditional expectation as a baseline value is approximately valid because feature independence approximately holds on a local sense, and although the models we study are highly non-linear, Assumption A1 guarantees a weaker sense of interpretability. However, when these two conditions are not satisfied, one should deploy different methods to approximate the conditional distribution as in Eq. (11). Lastly, note that our method relies on $\hat{f}$ satisfying A1, and one should wonder when this holds. Such an assumption is true when $f^*$–the true classification rule $Y = f^*(X)$–satisfies A1 (which is true for a variety of problems, including the ones studied in our experiments), and $\hat{f}$ constitutes a good approximation for $f^*$. As demonstrated in this work, such assumptions are reasonable in practical settings.

### 6.3 Multi-class extensions

Even though we have focused on binary classification tasks in this work, h-Shap can also be applied to multi-class settings. We now briefly demonstrate this by modifying the P. vivax experiment. We let $Y \in \mathcal{Y} = \{0, 1\}^2$, such that $Y = (\texttt{trophozoite}, \texttt{ring})$. Then, $\texttt{trophozoite} = 1$ if and only if there is at least one *trophozoite* in the image, and $\texttt{ring} = 1$ if and only if there is at least one *ring cell* in the image. Note that in this setting, these two classes are not mutually exclusive, as is typically the case for traditional image classifications problems. The latter setting is simply a particular case of the former. We randomly choose a training split that contains $80\%$ of each class, and we finetune a ResNet18 pretrained on ImageNet. We optimize all parameters for 60 epochs with Adam [43], using binary cross-entropy loss per class (as the classes are not mutually exclusive), learning rate of $0.0001$, weight decay of $0.00001$, and learning rate decay of $0.7$ every 7 epochs. After training, the model achieves an accuracy of $\approx 87\%$ on each label across the held-out test set. Fig. 6 shows saliency maps for two example images from the test set, one containing 6 trophozoites, and one containing 1 ring cell. h-Shap can explain every class, and it retrieves the desired, different types of cells. We regard studying the full implications and capabilities of h-shap in multi-class MIL problems as future work.



(a) $\texttt{trophozoite} = 1$. (b) Explanation for label `trophozoite`.

(c) $\texttt{ring} = 1$. (d) Explanation for label `ring`.

Fig. 6: Example saliency maps for different labels in a multiclass setting.

## 7 Conclusion

We presented a fast, scalable, and exact explanation method for image classification based on a hierarchical extension of Shapley coefficients. We showed that when the data distribution satisfies a multiple instance learning assumption, our method gains an exponential computational advantage while producing accurate–or approximate, if desired–results. Furthermore, we studied synthetic and real settings of varying complexity, demonstrating that h-Shap outperforms the current state-of-the-art methods in both accuracy and runtime, and suggesting that h-Shap acts as a weakly-supervised object detector. We have also presented and illustrated limitations of our approach, and addressing them is matter of future work.

## References

[1] C. Zednik, "Solving the black box problem: a normative framework for explainable artificial intelligence," *Philosophy & Technology*, pp. 1–24, 2019.

[2] R. Tomsett, D. Braines, D. Harborne, A. Preece, and S. Chakraborty, "Interpretable to whom? a role-based model for analyzing interpretable machine learning systems," *arXiv preprint arXiv:1806.07552*, 2018.

[3] J. Amann, A. Blasimme, E. Vayena, D. Frey, and V. I. Madai, "Explainability for artificial intelligence in healthcare: a multidisciplinary perspective," *BMC Medical Informatics and Decision Making*, vol. 20, no. 1, pp. 1–9, 2020.

[4] M. E. Kaminski, "The right to explanation, explained," *Berkeley Tech. LJ*, vol. 34, p. 189, 2019.

[5] M. E. Kaminski and G. Malgieri, "Algorithmic impact assessments under the gdpr: producing multi-layered explanations," *U of Colorado Law Legal Studies Research Paper*, no. 19-28, 2019.

[6] P. Hacker, R. Krestel, S. Grundmann, and F. Naumann, "Explainable ai under contract and tort law: legal incentives and technical challenges," *Artificial Intelligence and Law*, pp. 1–25, 2020.

[7] D. Shin, "The effects of explainability and causability on perception, trust, and acceptance: Implications for explainable ai," *International Journal of Human-Computer Studies*, vol. 146, p. 102551, 2021.

[8] M. T. Ribeiro, S. Singh, and C. 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*, 2016, pp. 1135–1144.

[9] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.

[10] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *International Conference on Machine Learning*. PMLR, 2017, pp. 3319–3328.

[11] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," in *International Conference on Machine Learning*. PMLR, 2017, pp. 3145–3153.

[12] S. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *arXiv preprint arXiv:1705.07874*, 2017.

[13] L. S. Shapley, "A value for n-person games," *Contributions to the Theory of Games*, vol. 2, no. 28, pp. 307–317, 1953.

[14] H. Shah, P. Jain, and P. Netrapalli, "Do input gradients highlight discriminative features?" *arXiv preprint arXiv:2102.12781*, 2021.

[15] I. Covert, S. Lundberg, and S.-I. Lee, "Explaining by removing: A unified framework for model explanation," *Journal of Machine Learning Research*, vol. 22, no. 209, pp. 1–90, 2021.

[16] J. Chen, L. Song, M. J. Wainwright, and M. I. Jordan, "L-shapley and c-shapley: Efficient model interpretation for structured data," *arXiv preprint arXiv:1808.02610*, 2018.

[17] J. MacDonald, S. Wäldchen, S. Hauch, and G. Kutyniok, "A rate-distortion framework for explaining neural network decisions," *arXiv preprint arXiv:1905.11092*, 2019.

[18] C. Heiß, R. Levie, C. Resnick, G. Kutyniok, and J. Bruna, "In-distribution interpretability for challenging modalities," *arXiv preprint arXiv:2007.00758*, 2020.

[19] L. Merrick and A. Taly, "The explanation game: Explaining machine learning models using shapley values," in *International Cross-Domain Conference for Machine Learning and Knowledge Extraction*. Springer, 2020, pp. 17–38.

[20] H. Chen, G. Zheng, and Y. Ji, "Generating hierarchical explanations on text classification via feature interaction detection," *arXiv preprint arXiv:2004.02015*, 2020.

[21] C. Singh, W. J. Murdoch, and B. Yu, "Hierarchical interpretations for neural network predictions," *arXiv preprint arXiv:1806.05337*, 2018.

[22] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez, "Solving the multiple instance problem with axis-parallel rectangles," *Artificial intelligence*, vol. 89, no. 1-2, pp. 31–71, 1997.

[23] M. Sundararajan and A. Najmi, "The many shapley values for model explanation," in *International Conference on Machine Learning*. PMLR, 2020, pp. 9269–9278.

[24] G. Owen, *Game Theory*, 3rd ed. Academic Press New York, 1995.

[25] H. Chen, S. Lundberg, and S.-I. Lee, "Explaining models by propagating shapley values of local components," in *Explainable AI in Healthcare and Medicine*. Springer, 2021, pp. 261–270.

[26] G. Owen, "Values of games with a priori unions," in *Mathematical economics and game theory*. Springer, 1977, pp. 76–88.

[27] S. López and M. Saboya, "On the relationship between shapley and owen values," *Central European Journal of Operations Research*, vol. 17, no. 4, p. 415, 2009.

[28] F. Huettner and M. Sunder, "Axiomatic arguments for decomposing goodness of fit according to shapley and owen values," *Electronic Journal of Statistics*, vol. 6, pp. 1239–1250, 2012.

[29] A. B. Khmelnitskaya and E. B. Yanovskaya, "Owen coalitional value without additivity axiom," *Mathematical Methods of Operations Research*, vol. 66, no. 2, pp. 255–261, 2007.

[30] K. Aas, M. Jullum, and A. Løland, "Explaining individual predictions when features are dependent: More accurate approximations to shapley values," *Artificial Intelligence*, p. 103502, 2021.

[31] C. Frye, I. Feige, and C. Rowat, "Asymmetric shapley values: incorporating causal knowledge into model-agnostic explainability," *arXiv preprint arXiv:1910.06358*, 2019.

[32] P. Sturmfels, S. Lundberg, and S.-I. Lee, "Visualizing the impact of feature attribution baselines," *Distill*, vol. 5, no. 1, p. e22, 2020.

[33] J. Haug, S. Zürn, P. El-Jiz, and G. Kasneci, "On baselines for local feature attributions," *arXiv preprint arXiv:2101.00905*, 2021.

[34] D. Janzing, L. Minorics, and P. Blöbaum, "Feature relevance quantification in explainable ai: A causal problem," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2020, pp. 2907–2916.

[35] N. Weidmann, E. Frank, and B. Pfahringer, "A two-level learning method for generalized multi-instance problems," in *European Conference on Machine Learning*. Springer, 2003, pp. 468–479.

[36] Z. Han, B. Wei, Y. Hong, T. Li, J. Cong, X. Zhu, H. Wei, and W. Zhang, "Accurate screening of covid-19 using attention-based deep 3d multiple instance learning," *IEEE transactions on medical imaging*, vol. 39, no. 8, pp. 2584–2594, 2020.

[37] N. Hashimoto, D. Fukushima, R. Koga, Y. Takagi, K. Ko, K. Kohno, M. Nakaguro, S. Nakamura, H. Hontani, and I. Takeuchi, "Multi-scale Domain-adversarial Multiple-instance CNN for Cancer Subtype Classification with Unannotated Histopathological Images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 6 2020.

[38] G. Fu, X. Nan, H. Liu, R. Y. Patel, P. R. Daga, Y. Chen, D. E. Wilkins, and R. J. Doerksen, "Implementation of multiple-instance learning in drug activity prediction," in *BMC bioinformatics*, vol. 13, no. 15. BioMed Central, 2012, pp. 1–12.

[39] U. Faigle and B. Peis, "A hierarchical model for cooperative games," in *International Symposium on Algorithmic Game Theory*. Springer, 2008, pp. 230–241.

[40] E. Algaba and R. van den Brink, "The shapley value and games with hierarchies," *Handbook of the Shapley Value*, p. 49, 2019.

[41] H. Chen, J. D. Janizek, S. Lundberg, and S.-I. Lee, "True to the model or true to the data?" *arXiv preprint arXiv:2006.16234*, 2020.

[42] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim, "Sanity checks for saliency maps," *arXiv preprint arXiv:1810.03292*, 2018.

[43] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[44] V. Ljosa, K. L. Sokolnicki, and A. E. Carpenter, "Annotated high-throughput microscopy image sets for validation." *Nature methods*, vol. 9, no. 7, pp. 637–637, 2012.

[45] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[46] M. B. Jensen, M. P. Philipsen, A. Møgelmose, T. B. Moeslund, and M. M. Trivedi, "Vision for looking at traffic lights: Issues, survey, and perspectives," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 7, pp. 1800–1815, 2016.

[47] M. P. Philipsen, M. B. Jensen, A. Møgelmose, T. B. Moeslund, and M. M. Trivedi, "Traffic light detection: A learning algorithm and evaluations on challenging dataset," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*. IEEE, 2015, pp. 2341–2345.

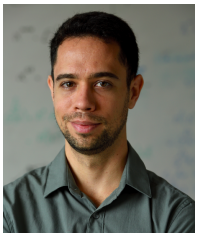
[48] R. Guidotti, "Evaluating local explanation methods on ground truth," *Artificial Intelligence*, vol. 291, p. 103428, 2021.

This article has been accepted for publication in IEEE Transactions on Pattern Analysis and Machine Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2022.3189849



**Jacopo Teneggi** received his B.S. in Biomedical Engineering from Politecnico di Torino, Italy, in 2020. He is a second-year Master's in Science and Engineering student at the Biomedical Engineering Department, Johns Hopkins University, and is affiliated with the Mathematical Institute for Data Science (MINDS). He hopes to pursue his PhD and focus on explainability and weakly supervised learning. He is passionate about machine learning, parmigiano reggiano, and aceto balsamico. Jacopo is a member of IEEE's HKN Honors Society.

**Alexandre Luster** received the B.S. degree in Life Sciences Engineering from the Swiss Federal Institute of Technology in Lausanne (EPFL), Switzerland, in 2020, where he is currently pursuing the M.S. degree in Life Sciences Engineering and Data Science. His research interests span computer vision, explainable artificial intelligence, precision medicine and neuroscience. He will conduct his Master's thesis in translational neuroscience at the Harvard Medical School.



**Jeremias Sulam** received his bioengineering degree from Universidad Nacional de Entre Ríos, Argentina, in 2013, and his PhD in Computer Science from the Technion – Israel Institute of Technology, in 2018. Since 2018, he is an assistant professor at the Biomedical Engineering Department, Johns Hopkins University, and affiliated with the Center for Imaging Science (CIS) and the Mathematical Institute for Data Science (MINDS). He is the recipient of the Best Graduates Award of the Argentinean National Academy of Engineering. His research interests include signal and image processing, sparse representation modeling, inverse problems and machine learning, and their application to biomedical problems. He is a member of the IEEE.