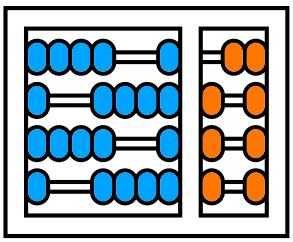
University of Campinas

Institute of Computing



Doctoral Qualifying Exam

Exploring Explaining Methods in Multi-Label Problems and Complementary Regularization Strategies in Weakly Supervised Semantic Segmentation

---

Candidate: Lucas Oliveira David

Advisor: Prof. Dr. Zanoni Dias

Co-advisor: Prof. Dr. Hélio Pedrini

Abstract

Over the years, many weakly supervised strategies have been devised to mitigate the necessity for large amounts of supervised annotation in segmentation tasks. As classification models can be conjointly employed with explaining methods to produce noisy segmentation proposals, weakly supervised strategies often rely on complex regularization techniques to instigate the development of useful properties (e.g., completeness, fidelity to semantic boundaries). In this work, we divide our contributions in two stages. In the former, we evaluate the efficacy of CAM-based techniques over distinct multi-label sets. We find that techniques that were created with single-label classification in mind (such as Grad-CAM, Grad-CAM++ and Score-CAM) will often produce diffuse visualization maps in multi-label scenarios, overstepping the boundaries of their explaining objects of interest onto objects of different classes. We propose a generalization of the Grad-CAM technique, namely MinMax-CAM, for the multi-label scenario that produces more focused explaining maps by maximizing the activation of a class of interest while minimizing the activation of the remaining classes present in the sample. We then propose a regularization strategy that encourages sparse positive weights in the last classifying, while penalizing the association between the classification of a class and the occurrence of correlated patterns, resulting in cleaner activation maps. Finally, we investigate complementary Weakly Supervised Semantic Segmentation techniques and regularizing strategies, discussing their strengths and limitations, and proposing direct extensions. Our preliminary results indicate MinMax-CAM produces more focused explaining maps over different network architectures and datasets, while our proposed approach to semantic segmentation substantially improves the effectiveness of three baselines without using additional training information or supervision.

1 Introduction

The adoption of Convolutional Neural Networks (CNNs) in the solution of a broad set of modern Machine Learning (ML) problems is unquestionable [44]. Today, we can easily find such models being employed to image classification [58], object detection [19] and localization [93], image segmentation [50], pose estimation [81] and even non-imagery domains, such

as audio processing [57], text classification [87] and text-to-speech [72].

In spite of their unquestionable efficacy, the extensive composition of complex operations in CNNs diminishes their overall interpretability, rendering "black box" models. As they gradually permeate into many real-world systems, impacting different demographics, the necessity for explaining and accountability becomes urgent. Scientists and engineers working with ML have since pushed towards the creation of explaining methods that could shed light into their inner workings [1, 9, 18, 41, 77, 90, 98].

Explaining the reasoning of an autonomous systems is challenging task, and yet paramount in increasing reliability of ML agents. While the construction of interpretable models is desirable as a general rule, as it facilitates the identification of failure modes while hinting strategies to fix them [62], it is also an essential component in building trust from the general public towards this technology [31].

In the context of Computer Vision (CV) and image classification problems, Explainable AI (XAI) can be employed to infer coarse localization cues that indicate the relative position of salient or class-specific visual patterns with respect to the visual receptive field. This property is frequently explored in the solution of Weakly Supervised Semantic Segmentation (WSSS) problems, making XAI methods a fundamental component in many WSSS methods.

Semantic Segmentation consists in correctly associating each pixel of an image or a video to a specific class from a predefined set, and is, to this day, one of the most prominent topics of study in CV [51]. Notwithstanding its complexity, it is a paramount component in any autonomous imagery reading system [6], such as self-driving vehicles [48], autonomous environment surveillance [26, 27], satellite imaging [91] and medical imagery [38, 83]. Representation Learning [5] solutions stand out in this task by consistently outscoring classic techniques across different areas, datasets and tasks [13]. However, these strategies often require massive amounts of densely annotated information (e.g., segmentation maps), obtained by extensive supervision. Considering limited time and cost constraints, these solutions remain inaccessible to many.

To circumvent these limitations, scientists and engineers often recur to Weakly Supervised Semantic Segmentation (WSSS) [55], where "weakly" refers to partially supervised

information, or lack thereof. Recent work investigated deriving semantic segmentation maps from saliency maps [32, 39, 43], bounding boxes [36, 40, 54], scribes and points [55], and even image-level labeled annotations [34, 37, 92]. Given it's similarity and shared goals, weakly supervised solutions are often compared to fully supervised ones, and while significant progress has been made so far, models trained in an weakly supervised setting often score significantly lower than the ones trained in a fully supervised manner.

1.1 Research Goals

We set forth the goal of studying Class-Specific explaining methods proposed so far in the multi-label setting, as well as developing a visualization technique which takes into account the expanded information available in multi-label problems. This study is important, from a scientific and engineering perspective, as it provides a comparison benchmark over more realistic scenarios, in which the capturing conditions are less controlled and more heterogeneous. Additionally, we remark the constantly increasing interest in Weakly Supervised Semantic Segmentation [7] and Localization [93] problems, in which XAI methods are frequently employed to extract localization cues, and class-specific precision is essential.

Subsequently, we aim to study promising weakly supervised regularization strategies (considering aspects such as efficacy, performance, applicability, and cross-influence) and to propose new extensions capable of further improving their individual and collective efficacy.

Finally, we intend to investigate the behavior of WSSS solutions to more complex boundary cases, such as long-tail and ambiguous functional segmentation problems. This investigation comprises a significant contribution to the understanding of weakly supervised problems, since many approaches proposed thus far focus on the concept of spatial visual affinity are strongly dependent on class frequency distribution and visual similarity.

1.2 Research Questions

In this section, we enumerate the questions that drive our research project. They are sorted according to the (expected) order in which they will be researched, and are further detailed in Section 4.

1. How do Explainable AI methods behave in multi-label scenarios, where metrics are computed for all occurring labels?

2. Can cross-contributions (non-discriminative contributions towards two or more classes) be erased from the Class Activation Maps produced by Grad-CAM?

3. What is the effect of context-decoupling methods over Semantic Segmentation of cluttered and dense scenes, where objects of distinct classes are presented close together?

4. How is overall efficacy of a WSSS system affected when complementary regularization strategies are conjointly employed? Can adversarial training improve the quality of the segmentation priors?

5. Can prediction ensembles from multiple WSSS strategies improve noisy segmentation priors? Can an ensemble policy for the selective employment of different strategies (conditioned to the problem and characteristics of the at hand), be learned to further improve overall effectiveness?

6. Can regularization strategies — originally proposed to reinforce segmentation-related properties in CNNs — be employed in the training of Transformer models with few or no modifications?

7. Can Visual Transformers be employed towards the improvement of fine-grain segmentation of small objects, containing complex non-convex semantic boundaries?

8. Can data balancing methods and long-tail learning be employed to further improve the efficacy of WSSS systems in extreme data unbalance settings?

9. How do modern WSSS methods — often relying on concepts such as pixel neighborhood similarity — fare on functional segmentation problems, in which semantic boundaries may not be clearly represented by visual cues?

10. Can the tasks of Saliency Detection and Semantic Segmentation be conjointly learned in a Weakly Supervised setting, promoting the improvement of overall efficacy by the use of mutual information between these tasks?

2 Theoretical Background

In this section, we enumerate and describe concepts that are essential to the understanding of our work.

Representation Learning A branch of Machine Learning concerned in learning useful data representations (along with the solution itself) for problems represented by unstructured samples or signals [5].

Semantic Segmentation A task that aims to obtain a segmentation of the elements in a signal with respect to their semantics. In Computer Vision, Semantic Segmentation often relates to associate each pixel to an element in a predefined set of classes [50].

Functional Segmentation The segmentation of elements composing a signal by their associated function or behavior, which may not be necessarily distinguishable by visual patterns or cues [7, 14].

Weakly Supervised Problems A ML paradigm that attempts to learn patterns from data with incomplete supervision or lack thereof, characterizing tasks or problems with noisy annotation and low human intervention [7]. Within the context of Semantic Segmentation, Weak Supervision often refers to the lack of manually constructed Semantic Segmentation annotation [55].

Long-Tail Class Distribution An extreme manifestation of class unbalance, in which classes are assigned to either *head* or *tail* sets. The *head* set has low cardinality, but contains classes that are well represented in the original set. Conversely, the *tail* set contains many classes that are sparsely represented [61]. Approaches to long-tail recognition vary from data re-sampling and re-weighting to the adoption of robust architectures [78, 97] and representation learning losses [96].

Convolutional Networks An ML learning model comprising convolution (or cross-correlation) operations, commonly employed in the task of Representation Learning (or Dimensionality Reduction) of Computer Vision problems represented by unstructured samples [29, 44, 90].

Striding and Dilation Properties that characterize the application of the discrete convolution over the spatial signal $x \in \mathbb{R}^{CHW}$. *Stride* refers to the sampling factor s of the passing signal [49] (i.e., the number of elements shifted when "sliding" the kernel during the convolution operation), implying in a reduction in the spatial dimensions of input signal. *Dilation* refers to the idea of convolving the original signal with a "spaced" kernel, containing "gaps" of size d between each element in its characteristic matrix [88]. The employment of either strategy entails in the convolution operation being applied over regions of gradually-increasing sizes, resulting in the expected stacking of patterns [5]. *Dilation*, however, has the advantage of maintaining the original resolution of the input signal (and the disadvantage of higher computational cost), being therefore frequently employed in networks devised for segmentation tasks [12, 82, 95].

Attention The means or capacity of a model to direct its focus towards the most informative portions of the data stream [30]. The "attention" provided by the model can segment the signal with respect to its spatial dimensions, its channels, or a combination thereof. The application of the first, spatial attention, creates a locally-connected (and spatially independent) system, whereas channel attention often results in the internalization of more robust set of data patterns [95].

Transformers A family of architectures based on attention and self-attention mechanisms [45]. Among them, Vision Transformers [21] (ViT) have been successfully applied to a broad range of weakly supervised visual tasks [25, 59, 60, 86].

3 Related Work

In this section, we discuss important landmarks reached in both XAI and WSSS literatures.

3.1 Explainable Artificial Intelligence in Computer Vision

Visual explanation techniques are frequently employed to describe or indicate, with a certain degree of certainty, salient cues that might have contributed to the decision process of

CNNs [85]. These techniques often times produce visual explaining maps: a signal with the same spatial format as the input sample, highlighting regions that most contribute to the answer provided by the model [75].

Gradient-based saliency methods [64] are early examples of this line of work. They produce saliency maps that highlight pixels with most overall contribution towards the score estimated during the decision process of a model, which is accomplished by back-propagating the gradient information from the units of interest, contained in the last layer, onto the input signal. Instances of these methods are Guided Backpropagation [66], which filters out the negative backpropagated gradients; SmoothGrad [65], which averages gradient maps obtained from multiple noisy copies of a single input image; and FullGrad [67], which combines the biases with the saliency information in order to create the "full gradient".

Notwithstanding their precision on locating salient regions and objects, gradient-based methods will ultimately fail to identify objects or regions associated with a specific class of interest. In fact, "sanity checks" have been proposed to test the resulting explaining maps from these methods when class-specific patterns are erased from the model. As examples, we remark the two experiments proposed by Adebayo et al. [1]: *Model Parameter Randomization* and *Data Randomization*. In the former, weights from layers would be progressively (or individually) randomized, from top to bottom, and the effect over the saliency map produced by each method would be observed. In the latter, labels would be permuted in the training set, forcing the network to memorize the noisy annotation. Some techniques, such as the Guided Backpropagation and Guided-CAM methods, were unaffected by the randomization of labels and weights of the top layers, demonstrating their invariance towards class information and high dependence on low-level features. These results lead the authors to conclude that those methods approximated the behavior of edge detectors.

Differently from gradient-based saliency methods, Class Activation Mapping (CAM) can be used to circumvent the lack of sensibility to class [98]. This technique consists in feed-forwarding an input image x over all convolutional layers of a CNN f and obtaining the positional activation signal $A^k = [a^k_{ij}]_{H \times W}$ for the k-th kernel in the last convolutional layer. If $W = [w^c_k]$ is the weight matrix of the last dense layer in f, then the importance of each

positional unit a_{ij} for the classification of label c is summarized as:

$$L^c_{\text{CAM}}(f, x) = \text{ReLU}(\sum_k w^c_k A^k) \tag{1}$$

Naturally, CAMs are not without shortcomings. Significant challenges ensue with the employment of CAMs: Firstly, only simple convolutional architectures can be explained through CAM, as it assumes a direct association between the activation convolutional signal and the classification signal. Additionally, when considering the later convolutional layers in the model, CAM will produce activation maps of considerably smaller size when compared to the input images. Hence, they must be upsampled (i.e., interpolated) to match their original counterparts, resulting in explaining maps with fairly imprecise object boundaries localization and highlighting. Furthermore, as the model focus on a few discriminative regions to predict a class for a given sample, the highlighted regions in the visualization map might not completely cover the salient objects associated with that specific class, being strongly affected by local patterns, the explaining method employed [9] and even the model's architecture [62]. In the context of visual explaining maps, this problem is strongly related to the concept of *prediction completeness* [74].

A broad spectrum of CAM-based methods have been developed in an attempt to to address the aforementioned problems and improve the quality of the explanations. Gradient signals were leveraged to extend CAM to Grad-CAM [62], in order to explain more complex network architectures, not limited to convolutional networks ending in simple layers such as Softmax classifiers and linear regression models. Let $S_c = f(x)_c$ be the score attributed by the network for class c with respect to the input image x, and $\frac{\partial S_c}{\partial A^k_{ij}}$ be the partial derivative of the score S_c with respect to the pixel (i, j) in the activation map A^k, then:

$$L^c_{\text{Grad-CAM}}(f, x) = \text{ReLU}\Big(\sum_k \sum_{ij} \frac{\partial S_c}{\partial A^k_{ij}} A^k\Big) \tag{2}$$

Chattopadhay et al. [9] then proposed Grad-CAM++ as an extension of Grad-CAM, in which each positional unit in A^k was weighted by leveling factors to produce maps that

evenly highlighted different parts of the image that positively contributed to the classification of class c, providing higher completeness for classes associated with large objects and multiple instances of the same object in the image [9]. Similarly to Grad-CAM, Grad-CAM++ is defined as:

$$L^c_{\text{Grad-CAM++}}(f,x) = \text{ReLU}\Big(\sum_k \sum_{ij} \alpha^{kc}_{ij} \text{ReLU}\Big(\frac{\partial S_c}{\partial A^k_{ij}}\Big) A^k\Big) \tag{3}$$

where

$$\alpha^{kc}_{ij} = \frac{\frac{\partial^2 S_c}{(\partial A^k_{ij})^2}}{2\frac{\partial^2 S_c}{(\partial A^k_{ij})^2} + \sum_{ab} A^k_{ab} \frac{\partial^3 S_c}{(\partial A^k_{ij})^3}}$$

The authors also proposed two new metrics: Increase of Confidence (%IC) and Average Drop (%AD), which have since been constantly employed in the evaluation of visual explaining methods.

Another visualization technique worth remarking is Score-CAM [77]. In it, visualization maps are defined as the sum of the activation signals A^k, weighted by factors C^k, that are directly proportional to the classification score obtained when the image pixels are masked by the normalized signal A^k. Formally:

$$L^c_{\text{Score-CAM}}(f,x) = \text{ReLU}\Big(\sum_k f(x \circ \frac{A^k}{\max A^k})_c A^k\Big) \tag{4}$$

More recently, an ever-growing interest in developing even more accurate visualization methods is noticeable. Among many, we remark SS-CAM [76], Ablation-CAM [18], Relevance-CAM [41], LayerCAM [33] and F-CAM [4]. Similarly to Score-CAM, Ablation-CAM is defined as the sum of feature maps A^k, where each map is weighted by the proportional drop in classification score when A^k is set to zero. Relevance-CAM combines the ideas of Grad-CAM with Contrastive Layer-wise Relevance Propagation (CLRP) to obtain a high resolution explaining map that is sensitive to the target class, while LayerCAM incorporates the signals advent from intermediate convolutional layers to increase the quality of explaining maps. Finally, F-CAM replaces the usual upscaling of the CAM by a parameterized reconstruction operation based on local statistics with respect to the objects of interest.

Notwithstanding the consistent progression towards the improvement of visualization results, the aforementioned methods entail significant computing footprint. We further note that much of the work conducted thus far have focused on evaluations over single-label multi-class datasets, such as localization task over ImageNet [62], and little investigation has been conducted over the effectiveness of these visualization techniques in multi-label scenarios. Additionally, studies that used multi-label datasets [9] often focus on single-label explanation (usually considering the highest scoring class as unit of interest).

As motivation, we present the visualization maps of classes of interest over a few samples from the Pascal VOC 2007 (VOC07) dataset [23] in Figure 1. In it, we observe a tendency of CAM-based methods (specially the most recent versions which attempt to expand the map to cover all parts of the classified object) to overflow the boundaries of the object of interest, even expanding over other objects of different classes.

3.2 Weakly Supervised Semantic Segmentation

WSSS is often approached as a two-stage process, in which the missing segmentation maps are derived from a weakly supervised dataset, and subsequently used as pseudo maps to train fully-supervised semantic segmentation models. Researchers in this area have focused on strategies comprising in (a) devising class-specific hints from coarse localization methods, such as Class Activation Mapping (CAM) [98], (b) encouraging *coverage completeness* by transferring label information from confident regions to a similar neighborhood; and (c) constraining segmentation proposals to boundaries of their associated objects [37].

The coarse maps can be refined by Random Walk (RW) [3] or Fully Connected Conditional Random Fields (CRF) [35]. In the former, the affinity values between pixel pairs that reside within a neighborhood are calculated, and used in a random walk procedure to extend labels from confident regions to uncertain ones. The authors later propose the addition of displacement fields in order to perform instance segmentation [2]. In the latter, unary and pairwise Gaussian potentials are used to model energy levels of each pixel, representing the confidence in its original label and its visual similarity to its neighborhood, respectively. Labels are reassigned to low confidence pixels with a high similarity to their neighborhood.

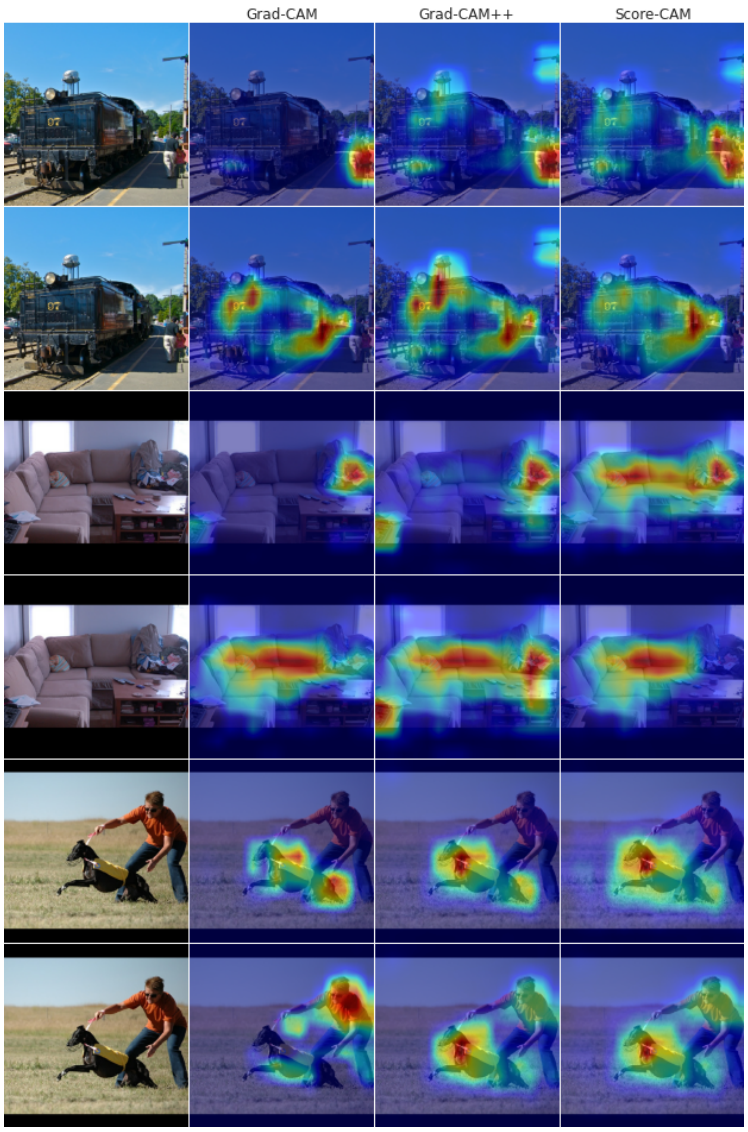


Figure 1: Explaining maps resulted from the application of various CAM-based visualization techniques over samples in the VOC07 dataset [23]. Source: David et al. [16]

Naturally, refinement methods are strongly affected by the prior seeds. Thus, many authors have focused on the development of strategies resulting in more accurate priors. In this vein, Jo and Yu proposed Puzzle-CAM [34] to reinforce *prediction completeness*. This is achieved by separating the input image according to its four quadrants, forwarding the four parts and reconstructing the output activation signal, resulting in a "local" information

stream. The model is trained to predict multi-label class occurrence within a sample via both main and local streams. The reconstructed signal, in turn, is used to regularize the main activation signal, resulting in CAMs with higher coverage over salient objects.

Let x_i be the i-th sample image in the training set, associated with the one-hot encoded vector $y_i \in \{0,1\}^c$, indicating the occurrence of at least one object associated with each one of the c existing classes in the set. At the same time, let f be a CNN such that $f^c(x_i) = A_i^c \in \mathbb{R}^{HW}$ is the spatial activation map with respect to sample x_i and class c, $\hat{A}_i^c = \text{merge}(f^c(\text{tile}(x_i)))$ the reconstruction of the tiled maps produced by separating x_i into four quadrants and forwarding them individually through f, and $p^c(A_i) = \sigma(\text{GAP}(A_i^c)) \in [0,1]$ the estimated posterior probability of sample x_i containing objects of class c. In these conditions, training ensures with the conjoint optimization of the following objective function:

$$\mathcal{L}_{\text{puzzle}}(x_i) = \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{p-cls}} + \alpha\mathcal{L}_{\text{re}} \tag{5}$$

where $\mathcal{L}_{\text{cls}}$ is the multi-label soft margin loss between $p(A_i)$ and y_i, $\mathcal{L}_{\text{p-cls}}$ is the multi-label soft margin loss between $p(\hat{A}_i)$ and y_i, $\mathcal{L}_{\text{re}} = \|A_i - \hat{A}_i\|_1$ is the *mean absolute error* loss between the main and (reconstructed) activation maps, and α is a scheduling coefficient that linearly increases as training progresses.

Notwithstanding its simplicity, Puzzle-CAM is reportedly associated with significantly high mIoU results over VOC12 dataset [34]. When employing their most successful model (using the split-attention architecture ResNeSt269 [95]), the authors obtained 71.9% and 72.2% mIoU over the validation and test subsets, respectively. However, upon closer inspection over the training loop, it becomes evident the existence of an *early stopping* mechanism that persists the weights of the model as training progresses, conditioned to the improvement of the metric of interest (mIoU)[1]. Hence, privileged and fully-supervised information (advent from the ground-truth maps) is being incorporated in the training procedure, further fortifying it against overfit, albeit mischaracterizing a supposedly weakly-supervised problem. It is also worth remarking that many of the WSSS studies conducted so far [2, 3, 37, 39, 43]

[1]Training procedures were made available by the authors on GitHub (accessed on Jan., 2023): (1) github.com/shjo-april/PuzzleCAM/train_classification_with_puzzle.py#L444-L448; (2) github.com/shjo-april/PuzzleCAM/train_segmentation.py#L333-L336

have not employed similar early stopping or weight persistence mechanisms, implying that these results cannot be directly compared.

Another interesting approach devised to generate better segmentation priors is Class-specific Adversarial Erasing (CSE) [37]. It consists of an assisted training setup, in which a class r_i is randomly drawn from the set of labels associated with the i-th sample, and a CAM Generating model f (namely CGNet) is fitted to produce activation maps that, when masking the input images, minimize the classification output signal advent from a fixed auxiliary ordinary classifier (OC), while maintaining the original output for the remaining classes. In practice, CAMs learned in this setup become sufficiently accurate to insulate objects of distinct classes, increasing coverage over their objects while maintaining coarse fidelity to semantic boundaries.

Let f be the CAM proposal (main) network such that $f^c(x_i) = A_i^c \in \mathbb{R}^{HW}$ and oc be the ordinary, fixed, classifying network. For each pair (x_i, y_i) in the training set, a class d is randomly sampled from y_i. In these conditions, Class-specific Adversarial Erasing [37] is defined as the minimization of the function:

$$\begin{aligned} \mathcal{L}_{\text{cse}}(x_i) &= \mathcal{L}_{\text{cls}} + \alpha \mathcal{L}_{\text{cae}} \\ &= \ell_{\text{bce}}(\text{GAP}(A_i), y_i) \\ &+ \alpha \ell_{\text{bce}}(oc(x_i \circ (1 - A_i^d)), y_i \setminus \{d\}) \end{aligned} \tag{6}$$

where ℓ_{bce} is the binary cross-entropy loss function.

Once trained, f can be used to devise segmentation priors, which are refined with Random Walk [3] and CRF [35], resulting in pseudo segmentation maps. A DeepLab model [11], trained over these same maps, obtains 68.4% and 68.2% mIoU over Pascal VOC 2012 validation and testing sets, respectively, and 36.4% over the MS COCO 2014 validation set [46].

Notwithstanding the noticeable improvement in class separation and fidelity to the semantic boundaries, priors from OC-CSE still display a low coverage over salient objects. This drawback is mitigated by the authors with the employment of Random Walk. Naturally, the effectiveness of CSE is strongly dependent on the capacity of oc to recognize

class-specific objects (and parts thereof), and it can be potentially diminished when assisted by a classifier biased towards (only) the most discriminative regions. It stands to reason that the concurrent training of *oc* could prove itself useful to the CSE method, in which the ordinary classifier would gradually learn to redirect its attention to class-specific regions ignored currently by f, and thus providing better assistance in its training.

Going in a different direction, Xie et al. proposed C^{2}AM: an unsupervised strategy for learning saliency detection [84]. It attempts to find a bi-partition of the spatial field containing the image such that salient objects and the background would be perfectly separated. It does so by extracting both low and high level features $A^k \in \mathbb{R}^{HW}$ from a pretrained backbone model, and feeding them to a *disentangling* branch, a function $d : \mathbb{R}^K \to [0, 1]$, such that $d(A)A$ would represent the *foreground* features while $(1-d(A))A$ represented the *background* ones. Training ensues by optimizing the model to approximate the feature vectors representing the most similar patches, while increasing the distance between the *foreground* and *background* feature vectors. Saliency maps produced by C^{2}AM can be combined with various WSSS strategies, notably improving their effectiveness.

Let $P^{hw} : \mathbb{R}^{hw} \to [0, 1]^{chw}$ be a function mapping each region in the embedded spatial signal A_i^{hw} to the probability value p_i^{hw} of said region belonging to the first partition. Moreover, let $v_i^f = P_i \circ A_i$ and $v_i^b = (1-P_i) \circ A_i$ be two extracted feature vectors, representing the spatial *foreground* (*fg*) and *background* (*bg*) features, respectively. Considering a batch of n images $\mathcal{B} = \{x_b, x_{b+1}, \ldots, x_{b+n-1}\}$, three cosine similarity matrices are calculated: (a) the *fg* features (s_{ij}^f), (b) the *bg* features (s_{ij}^b); and (c) between the *fg* and *bg* (s_{ij}^{neg}) features. In these conditions, C^{2}AM is defined as the minimization of the following objectives:

$$
\begin{aligned}
\mathcal{L}^{\mathcal{B}}_{\text{C}^2\text{AM}} &= \mathcal{L}^{\mathcal{B}}_{\text{pos-f}} + \mathcal{L}^{\mathcal{B}}_{\text{pos-b}} + \mathcal{L}^{\mathcal{B}}_{\text{neg}} \\
&= \frac{1}{n(n-1)} \sum_i^n \sum_j^n \mathbb{1}_{[i \neq j]} (w_{ij}^f \log s_{ij}^f) \\
&+ \frac{1}{n(n-1)} \sum_i^n \sum_j^n \mathbb{1}_{[i \neq j]} (w_{ij}^b \log s_{ij}^b) \\
&- \frac{1}{n^2} \sum_i \sum_j \log(1 - s_{ij}^{\text{neg}})
\end{aligned}
\tag{7}
$$

where the w_{ij}^f and w_{ij}^b factors are exponentially proportional to the similarity rank between the regions (i, j), considering all possible pairs available in A_b:

$$w_{ij} = e^{-\alpha \text{rank}(s_{ij})}, w \in \{w^f, w^b\}$$

Once trained, pseudo saliency maps are inferred from the training set, and used to train a fully-supervised saliency detection model [47]. The model, in turn, is used to generate the saliency proposal maps, which tend to be more robust to noise than the pseudo maps. Concatenating said maps to their respective CAMs comprises a way to perform pixel-wise thresholding, which tends to point out background regions more precisely than establishing a global threshold value. For instance, the combination of C^{2}AMs to CAMs produced by ResNet50 Puzzle resulted in a 65.5% mIoU over the VOC12 training set (an increase of 14.1 percent points), and in 66.0% mIoU when combined with priors produced by a model when employing SC-CAM [8]. Results over the validation set were not provided.

C^{2}AM is not without shortcomings. With careful inspection of Eq. (7), it is noticeable the absence of an "anchor": similar pixels representative of salient objects (relative to the dataset of interest) can either be associated with low or high values in P^{hw}. I.e., C^{2}AM establishes a saliency bi-partition of the visual receptive field, without specifying which of the partitions contains the salient objects. Moreover, no explicit reinforcement is made towards the construction of a bi-partition that aggregates all salient classes in one side. Instead, similar regions are simply drawn together, implying in the risk of salient objects, associated with different classes, to be projected onto different partitions. For example, in problems where objects of two classes never directly co-occur (spatially close in a same image), or indirectly (through a third intermediate class that frequently co-occurs with each of the aforementioned classes).

4 Research Methodology, Materials and Contributions

In this section, we describe our proposed research approach, as well as the contributions achieved thus far.

4.1 Proposed Approach

In the following, we enumerate the different stages of our proposed approach, sorted by the order in which they will be studied and researched.

4.1.1 Exploration of Explainable AI Methods in Multi-Label Problems

We propose to start our project by conducting a throughout evaluation of the main CAM-based visual explaining techniques and methods, considering the neglected aspects of multi-label problems and scenarios, in which analysis can be considerably more challenging [73].

We will extend the well established XAI metrics defined by Chattopadhay et al. [9] to consider the multiple class occurrences in each sample. In this benchmark, the explaining methods should not only be evaluated with respect to their capacity of explaining a single class of interest, but to with respect to its capacity of explaining all salient elements currently present in samples.

Moreover, we will include additional problem sets in the evaluation procedure, representing more complex and realistic scenarios not currently covered by the well-established (and well curated) Pascal VOC and MS COCO datasets. We argue that the evaluation over those is paramount to better estimate the effectiveness of XAI methods over more realistic scenarios, containing degeneration cases, such as when the classes are sparsely represented, class distribution is strongly unbalanced or with extreme class co-occurrence [7].

Finally, we will devise a new CAM-based visual explaining method that takes into consideration the various classes present in a sample to retrieve the kernel regions (i.e., regions that contribute to the recognition of one, and only one, class of interest) in that same sample. This will be achieved through the contrasting the contributions of the model for the prediction of each class. For fairness, the devised method will be compared against the literature using the aforementioned comprehensive evaluation loop.

4.1.2 Complementary Regularization Strategies in WSSS

In this research stage, we will evaluate the efficacy of complementary regularization strategies, devised in the context of noisy or weak supervision (e.g., Puzzle-CAM, OC-CSE, C^2AM,

label smoothing, etc.), and strong augmentation strategies devised to internalization of more robust data patterns in an uncertain and noisy environment (e.g., CowMix, ClassMix, CertainMix). We expect that revisiting these individual solutions may prove itself useful in the understanding and development of new techniques that retain their individual strengths without suffering from their shortcomings.

Finally, we will devise a new WSSS strategy that utilizes an adversarial training setup of two CAM-proposing networks to produce more accurate pseudo semantic segmentation priors, further improving the overall efficacy of WSSS solutions.

4.1.3 Exploration of Transformers and Spatial Attention for Highly-Detailed Segmentation

Modern WSSS solutions are often based on CNNs, comprising convolution and downsampling operations that compress the original spatial dimensions of the input signal, resulting in low-resolution semantic segmentation priors. While some work has been conducted to mitigate this flaw, such as the proposal of wide networks [82] and the introduction of dilation [12,88], the segmentation of small objects with highly detailed non-convex semantic boundaries is still challenging.

Alternatively to CNNs, Vision Transformers [21] can also be employment in the solution of various Computer Vision tasks, achieving or surpassing the state of the art in many of them. Concomitantly, spatial attention can be used to maintain a higher fidelity to the original resolution of the analyzed sample, resulting in more accurate segmentation priors. Early work in this vein have inferred class affinity from patch tokens advent from Transformers [86], employed linear search to assign image-level class information to patches [59], or combined CNNs and Transformers into single-stage multi-branch model with classification, spatial affinity prediction and segmentation capabilities [60].

In this research stage, we will analyze the effectiveness of modern Vision Transformers architectures over Weakly Supervised Semantic Segmentation problems. More specifically, we will investigate if modern regularization strategies, originally devised to instigate segmentation-prone properties in CNNs (e.g., completeness, local attention, activation

consistency, semantic boundaries), can be extended and adapted to improve the semantic segmentation capabilities of Transformer models, without the overhead of adjacent convolutional layers/models that would inevitably lead to complex architecture topologies.

4.1.4 Weak Supervision in Boundary and Difficult Scenarios: Class Unbalance, Long-tail and Functional Segmentation

In this phase, we will investigate the behavior of WSSS strategies in boundary and difficult scenarios, such as in datasets with unbalanced or long-tail class distributions. We expect the segmentation capacity of a model to deteriorate in poorly represented scenarios, leading to a significant difference in segmentation effectiveness across the different class groups.

We will study ways to mitigate this problem by leveraging re-balancing and class-influence adjustment techniques — originally devised with classification and recognition tasks in mind — and evaluating their influence on the quality of the semantic segmentation priors produced in a weakly supervised environment. More specifically, we intent to adapt the distribution alignment [96] and the Bilateral-branch [97] methods to the WSSS scenario, producing high-quality semantic segmentation priors for both *head* and *tail* classes.

Subsequently, we will investigate the effectiveness of weakly supervised techniques when applied to functional and morphological segmentation tasks. We expect solutions based on pixel-wise visual affinity to present considerably lower performance in scenarios containing visually ambiguous patterns, in which the fully-supervised information cannot be solely inferred from local features. Finally, we will attempt to mitigate these scenarios using more interventionist instances of weakly supervised annotation (e.g., scribes, bounding boxes, and saliency), or by adapting semi-supervised learning methods to this problem domain.

4.1.5 Ensemble of Weakly Supervised Semantic Segmentation Systems

While many WSSS solutions have similar overall efficacy (mIoU), their fundamentally different architectures and training objectives culminate in models with different segmentation capacity with respect to different groups of classes, as well as a diverse set of failure cases. In this scenario, it is unlikely for a single model to have the highest segmentation capability

among all groups, contexts and problems.

Inspired by the continuous success of prediction ensembling in various tasks and competitions [20], as well as the recent adoption of model and weight ensembling [79], we intent to analyze the efficacy of ensembling predictions advent from various WSSS methods towards the solution of semantic segmentation tasks.

Finally, we will devise a meta learning strategy that combines different WSSS methods and techniques based on any contextual and/or weakly supervised information available, favoring methods that better perform considering the current context being inferred. We expect an organized composition of predictions to produce more robust semantic segmentation results in all scenarios, including boundary and exceptional cases.

4.2 Experimental Environment and Materials

In this section, we detail the experimental setup employed in our work.

4.2.1 Datasets

We list and briefly describe the datasets considered for this work. The first five datasets have been employed in the experiments and are thus discussed in more depth in Appendix A.1.1.

Pascal VOC 2007 comprises 2,501 training samples, 2,510 validation samples and 4,952 test samples representing various objects from 20 classes in their usual context [23].

Pascal VOC 2012 extends the 2007 version to include 5,717 training samples, 5,823 validation samples and 10,991 unlabeled test samples [22].

MS COCO 2017 contains 118,287 training samples, 5,000 validation samples and 40,670 unlabeled test samples. This set represents distinct scenarios containing various objects associated to 80 distinct classes [46].

Planet: Understanding the Amazon from Space is a satellite imagery dataset of the Amazon rainforest, containing 40,479 training samples annotated according to their natural features [63].

Human Protein Atlas Image Classification (HPA) is a microscopic imagery dataset representing cellular bodies and proteins of interest. Containing 31,072 training samples associated with one or more of the 21 classes, this set presents a strongly unbalanced class distribution [56].

Functional Map of the World (fMoW) is a satellite imagery dataset 1,047,691 images associated with 61 categories [14]. Samples comprise temporal sequences of images, annotated by bounding boxes associated with 61 distinct categories that describe the functional purpose and/or contextual information of the scenarios.

Atlas of Digital Pathology (ADP) is a histopathology dataset containing 17,668 images, captured from histological tissue slides [28]. Images are annotated according to 28 morphological types and 4 functional types. Moreover, a small subset (of 50 images) presents fully-supervised pixel-level annotations.

4.2.2 Metrics and Evaluation Protocols

To evaluate Explainable AI methods, we extend the well known *Increase in Confidence* and *Average Drop* metrics to a multi-label scenario by computing them individually, for each label present in each sample, followed by macro-averaging the individual results. We then devise two new metrics, namely *Average Drop of Others* and *Average Retention*, to measure the inadvertent class-agnostic highlighting of co-occurring classes in CAMs. Finally, we summarize the devised metrics into the harmonic means F_1+ and F_1-.

In conformity with literature, we employ mean Intersection over Union (mIoU) as main evaluation metric when comparing Weakly Supervised Semantic Segmentation solutions.

Each of the aforementioned metrics are described in detail in Appendix A.1.3.

4.2.3 Computational Environment

Training of classification models, as well as the experiments and benchmarks of Explainable Artificial Intelligence methods are conducted in a local environment, consisting of a single node with 16 GB RAM and a NVIDIA T4 GPU.

We leverage the infrastructure of the Santos Dumont (SDumont) super-computer for the execution of the experiments associated with WSSS problems. Most experiments occur in single-node machines with 128 GB RAM and 4 NVIDIA V100 GPUs.

Training protocols for XAI and WSSS methods are described in detail in Appendix A.1.2 and Appendix A.2, respectively.

4.3 Work Schedule

We enumerate in this section the different research stages of our project, and present the execution scheduling for the planned activities in Table 1.

1. Class attendance and completion of required credits.
2. Exploration of Explainable AI methods in multi-label problems.
3. Complementary regularization strategies in WSSS.
4. Doctoral Qualifying Exam (EQE).
5. Participation in "Programa de Estágio Docente" (PED).
6. Exploration of Transformers and Spatial Attention for highly-detailed segmentation.
7. Weak supervision in boundary and difficult scenarios.
8. Ensemble of weakly supervised semantic segmentation systems.
9. Writing and presentation of Doctoral thesis.

Table 1: Expected scheduling of planned activities.

Activities	**1st year**				**2nd year**				**3rd year**				**4th year**			
	1	**2**	**3**	**4**	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**
1	•	•	•	•												
2	•	•	•	•												
3					•	•	•	•								
4									•							
5									•	•						
6									•	•	•					
7											•	•	•	•		
8														•	•	•
9																•

5 Preliminary Results

Our main contributions, achieved thus far, are summarized by the following:

1. We propose a thoroughly analysis of popular visualization techniques in the literature over a distinct set of multi-label problems, evaluating their results according to the offered coverage over objects belonging to the label of interest, as well as the containment within objects of said label.

2. We propose a modification to CAM-based methods that combines gradient information from multiple labels within a single input image. We demonstrate that our approach presents better scores and cleaner visualization maps than other methods over distinct datasets and architectures. Subsequently, we present a regularization strategy that encourages networks to associate each learned class with a distinct set of patterns, resulting in better separation of concepts and cleaner CAM visualizations.

3. We propose a training procedure regularized by both Puzzle-CAM [34] and OC-CSE [37] to generate attention maps that comprehensively cover large objects while respecting their semantic boundaries. Sub-sequentially, we extend OC-CSE into a fully adversarial training setup, in which the ordinary classifier is also gradually refined to provider better information to the main network.

4. We propose an extension of C^2AM [84] that incorporates hints of positive regions in its training procedure, and empirically demonstrate the superiority of the generated saliency maps with respect to the ones obtained from C^2AM vanilla. We leverage the obtained pseudo saliency maps to infer background regions and better guide the random walk process [3], resulting in superior *mean Intersection over Union* (mIoU) results over the Pascal VOC 2012 (VOC12) dataset [22].

5.1 Explainable Artificial Intelligence

We devise a new CAM-based visual explaining method, namely MinMax-CAM, that produces more focused explaining maps by contrasting the contributions to the classification of

all classes contained in a given sample. Next, we briefly describe the method, and present a more detailed explanation of its intuition and formulation in Appendix B.1.

Formally, let f be a fully convolutional network, $x \in \mathcal{X}$ be a given sample from the dataset $\mathcal{X}$, $c \in C_x$ a class of interest present in the label set C_x, and $N_x = C_x \setminus \{c\}$. In these conditions, $S_c = f(x)_c = \sum_k w_k^c \frac{1}{hw}(A^k)$ is the prediction score for sample x with respect to class c, estimated by f, and MinMax-Grad-CAM is defined as the combination of activation signals A^k, weighted by their respective contributions to the objective function J_c:

$$L^c_{\text{MinMax-Grad-CAM}}(f, x) = \text{ReLU}\Big(\sum_k \sum_i \frac{\partial J_c}{\partial A^k_{ij}} A^k\Big) \tag{8}$$

where

$$J_c = S_c - \frac{1}{|N_x|} \sum_{n \in N_x} S_n \tag{9}$$

An alternative form (D-MinMax-Grad-CAM), that factors *positive*, *negative* and *background* contributions is also devised:

$$L^c_{\text{D-MinMax-Grad-CAM}} = \text{ReLU}\Big(\sum_k \alpha^c_k A^k\Big) \tag{10}$$

where

$$\alpha^c_k = \sum_{ij} \left[\text{ReLU}\Big(\frac{\partial S_c}{\partial A^k_{ij}}\Big) - \frac{1}{|N_x|}\text{ReLU}\Big(\sum_{n \in N_x} \frac{\partial S_n}{\partial A^k_{ij}}\Big) + \frac{1}{|C_x|} \min\Big(0, \sum_{n \in C_x} \frac{\partial S_n}{\partial A^k_{ij}}\Big)\right] \tag{11}$$

Qualitative results can be inspected in Figure 2, in which CAMs devised from MinMax-CAM and D-MinMax-CAM are illustrated in the fifth and sixth columns, respectively. By suppressing the activation of regions that positively contribute to the classification of adjacent classes, MinMax-CAM produces more precise and class-specific activation maps, in which fewer pixels (associated with a certain class) are incorrectly highlighted (when explaining another class).

More qualitative and quantitative results are presented in detail in Appendix B.1.2.

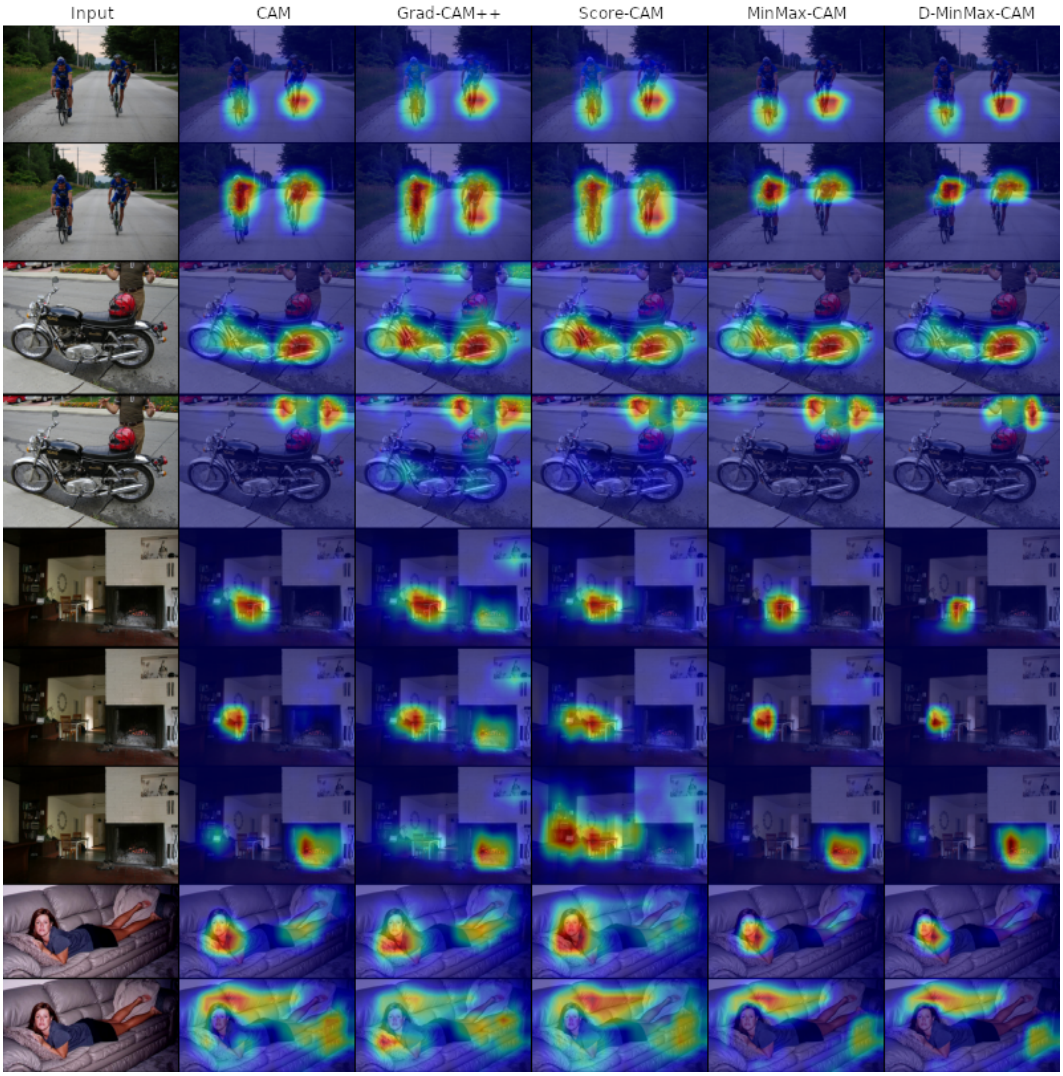


Figure 2: CAMs produced by various XAI methods. Classes being explained are (from top to bottom): *bicycle*, *person*, *motorbike*, *person*, *table*, *chair*, *tv*, *person*, and *sofa*.

5.2 Weakly Supervised Semantic Segmentation

In order to foment semantic segmentation-like properties in models trained in a WSSS setup, such as *prediction completeness*, semantic boundary awareness and robustness against noisy labels, we propose the combination of the Puzzle-CAM, OC-CSE and *label smoothing* strategies into a single training setup, namely P-OC.

Furthermore, we propose a novel adversarial training setup (namely, P-NOC), in which the Ordinary Classifier is gradually refined to shift its attention to regions being currently ignored by the main network, maximizing the utility in the regularization of the latter.

Finally, we propose the utilization of *saliency hints*, extracted from models trained in the weakly supervised scheme, to further regularize the training of C^{2}AM. This is accomplished by extracting high-confidence salient regions from the semantic segmentation priors and utilizing them to "anchor" the disentangling branch in the C^{2}AM model.

Table 2 displays the comparison with the state of the art for our two best strategies (P-OC and P-NOC). P-NOC obtains 72.7% mIoU over Pascal VOC 2012 test dataset, outscoring the unfair version of Puzzle-CAM by 0.5 p.p., and the remaining approaches by a considerable margin. Figure 3 and Figure 3 illustrate a few examples of predictions made by the P-OC and P-NOC strategies, respectively.

Figure 3: Examples of predictions made by a DeepLabV3+ model trained with pseudo semantic segmentation masks devised from P-OC and refined with random walk.

Implementation details for both P-OC, P-NOC and C^{2}AM-H are available in Appendix B.2, and quantitative results and ablation studies are described in Appendix B.2.3.

Figure 4: Examples of predictions made by a DeepLabV3+ model trained with pseudo semantic segmentation masks devised from P-NOC and refined with random walk.

Table 2: Comparison with other SOTA methods. mIoU (%) scores are reported for both Pascal VOC 2012 validation and testing sets. Puzzle-CAM: potential effectiveness reported (see Section 3.2).

Method	**Backbone**	**Val**	**Test**
AffinityNet [3]	Wide-ResNet-38	61.7	63.7
IRNet [2]	ResNet-50	63.5	64.8
ICD [24]	ResNet-101	64.1	64.3
SEAM [80]	Wide-ResNet-38	64.5	65.7
OC-CSE [37]	Wide-ResNet-38	68.4	68.2
Puzzle-CAM [34]	ResNeSt-269	71.9	72.2
RIB [39]	ResNet-101	68.3	68.6
EPS [43]	ResNet-101	70.9	70.8
AMN [42]	ResNet-101	69.5	69.6
ViT-PCM [59]	ViT-B/16	70.3	70.9
MCTformer [86]	Wide-ResNet-38	**71.9**	71.6
P-OC$_{+\text{C}^2\text{AM-H}}$ (ours)	ResNeSt-269	71.4	72.4
P-NOC$_{+\text{LS}+\text{C}^2\text{AM-H}}$ (ours)	ResNeSt-269	71.5	**72.7**

5.3 Scientific Production

1. L. David, H. Pedrini, and Z. Dias. MinMax-CAM: Improving focus of CAM-based visualization techniques in multi-label problems. In *17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP)*, pages 106–117. INSTICC, SciTePress, 2022.

2. L. David, H. Pedrini, and Z. Dias. MinMax-CAM: Increasing Precision of Explaining Maps by Contrasting Gradient Signals and Regularizing Kernel Usage (Springer).

In *17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP)*, CCIS Series, 2023.

3. L. David, H. Pedrini, and Z. Dias. Not so Ordinary Classifier: Revisiting Complementary Regularizing Strategies for More Robust Priors in Weakly Supervised Semantic Segmentation.

5.4 Technical Contributions

1. Implement pixel ignoring functionality in the cross-entropy loss in Keras, for semantic segmentation problems[2].

2. Ported the Wide ResNet38-d and ResNeSt architectures, originally trained in PyTorch, to TensorFlow[3].

3. Created the `keras-explainable` library, containing out-of-the box implementations of many Explainable AI algorithms[4].

4. Various fixes in Keras and TensorFlow-Addons, often related to the optimizer, mixed-precision when training in a Multi-Worker-Mirrored-Strategy environment[5].

Acknowledgments

The authors would like to thank CNPq (grant 140929/2021-5) and LNCC/MCTI (grant 46478) for providing HPC resources of the Santos Dumont (SDumont) supercomputer.

[2]Contribution logs available at: github.com/keras-team/keras/pull/16712.
[3]Available at: github.com/lucasdavid/resnet38d-tf and github.com/lucasdavid/resnest-tf.
[4]Available at: github.com/lucasdavid/keras-explainable.
[5]Contribution logs available at: #16664, #16922 #2714, #16332

References

[1] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim. Sanity checks for saliency maps. In *32nd International Conference on Neural Information Processing Systems (NIPS)*, page 9525–9536, Red Hook, NY, USA, 2018. Curran Associates Inc. 2, 7

[2] J. Ahn, S. Cho, and S. Kwak. Weakly supervised learning of instance segmentation with inter-pixel relations. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2209–2218, 2019. 10, 12, 26

[3] J. Ahn and S. Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4981–4990, 2018. 10, 12, 13, 22, 26, 39, 59

[4] S. Belharbi, A. Sarraf, M. Pedersoli, I. Ben Ayed, L. McCaffrey, and E. Granger. F-CAM: Full resolution class activation maps via guided parametric upscaling. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 3490–3499, 2022. 9

[5] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013. 2, 5, 6

[6] B. Bhanu and S. Lee. *Genetic Learning for Adaptive Image Segmentation*. The Springer International Series in Engineering and Computer Science. Springer US, 2012. 2

[7] L. Chan, M. S. Hosseini, and K. N. Plataniotis. A comprehensive analysis of weakly-supervised semantic segmentation in different image domains. *International Journal of Computer Vision*, 129(2):361–384, 2021. 3, 5, 16, 40, 47, 48, 49

[8] Y.-T. Chang, Q. Wang, W.-C. Hung, R. Piramuthu, Y.-H. Tsai, and M.-H. Yang. Weakly-supervised semantic segmentation via sub-category exploration. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8991–9000, 2020. 15

[9] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian. Grad-CAM++: Generalized gradient-based visual explanations for deep convolutional networks. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 839–847. IEEE, 2018. 2, 8, 9, 10, 16, 34, 36

[10] A. Chaudhry, P. K. Dokania, and P. H. Torr. Discovering class-specific pixels for weakly-supervised semantic segmentation. *arXiv preprint arXiv:1707.05821*, 2017. 58

[11] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected CRFs. *arXiv preprint arXiv:1412.7062*, 2014. 13

[12] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *European Conference on Computer Vision (ECCV)*, pages 801–818, 2018. 6, 17, 65

[13] Y. Chen, J. Tao, L. Liu, J. Xiong, R. Xia, J. Xie, Q. Zhang, and K. Yang. Research of improving semantic image segmentation based on a feature fusion model. *Journal of Ambient Intelligence and Humanized Computing*, pages 1–13, 2020. 2

[14] G. Christie, N. Fendley, J. Wilson, and R. Mukherjee. Functional map of the world. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6172–6180, 2018. 5, 20

[15] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le. Randaugment: Practical automated data augmentation with a reduced search space. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 702–703, 2020. 39

[16] L. David., H. Pedrini., and Z. Dias. MinMax-CAM: Improving focus of cam-based visualization techniques in multi-label problems. In *17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 4: VISAPP*, pages 106–117. INSTICC, SciTePress, 2022. 11, 44, 45, 46, 48, 49, 52, 56

[17] I. Demir, K. Koperski, D. Lindenbaum, G. Pang, J. Huang, S. Basu, F. Hughes, D. Tuia, and R. Raskar. DeepGlobe 2018: A challenge to parse the earth through satellite images. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 172–181, 2018. 49

[18] S. Desai and H. G. Ramaswamy. Ablation-CAM: Visual explanations for deep convolutional network via gradient-free localization. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 983–991, 2020. 2, 9

[19] A. Dhillon and G. K. Verma. Convolutional neural network: a review of models, methodologies and applications to object detection. *Progress in Artificial Intelligence*, 9(2):85–112, 2020. 1

[20] X. Dong, Z. Yu, W. Cao, Y. Shi, and Q. Ma. A survey on ensemble learning. *Frontiers of Computer Science*, 14:241–258, 2020. 19

[21] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 6, 17

[22] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136, 2015. 19, 22, 35

[23] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The Pascal Visual Object classes (VOC) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010. 10, 11, 19, 34, 35, 41

[24] J. Fan, Z. Zhang, C. Song, and T. Tan. Learning integral objects with intra-class discriminator for weakly-supervised semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4283–4292, 2020. 26

[25] W. Gao, F. Wan, X. Pan, Z. Peng, Q. Tian, Z. Han, B. Zhou, and Q. Ye. Ts-cam: Token semantic coupled attention map for weakly supervised object localization. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2886–2895, 2021. 6

[26] F. Gelana and A. Yadav. Firearm detection from surveillance cameras using image processing and machine learning techniques. In *Smart Innovations in Communication and Computational Sciences*, pages 25–34. Springer, 2019. 2

[27] J.-J. Hernandez-Lopez, A.-L. Quintanilla-Olvera, J.-L. López-Ramírez, F.-J. Rangel-Butanda, M.-A. Ibarra-Manzano, and D.-L. Almanza-Ojeda. Detecting objects using color and depth segmentation with kinect sensor. *Procedia Technology*, 3:196–204, 2012. 2

[28] M. S. Hosseini, L. Chan, G. Tse, M. Tang, J. Deng, S. Norouzi, C. Rowsell, K. N. Plataniotis, and S. Damaskinos. Atlas of digital pathology: A generalized hierarchical histological tissue type-annotated database for deep learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11747–11756, 2019. 20

[29] X. Hou, L. Shen, K. Sun, and G. Qiu. Deep feature consistent variational autoencoder. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1133–1141. IEEE, 2017. 5

[30] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7132–7141, 2018. 6

[31] D. T. Huff, A. J. Weisman, and R. Jeraj. Interpretation and visualization techniques for deep learning models in medical imaging. *Physics in Medicine & Biology*, 66(4):04TR01, 2021. 2

[32] P.-T. Jiang, Y. Yang, Q. Hou, and Y. Wei. L2g: A simple local-to-global knowledge transfer framework for weakly supervised semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16886–16896, 2022. 3, 58

[33] P.-T. Jiang, C.-B. Zhang, Q. Hou, M.-M. Cheng, and Y. Wei. LayerCAM: Exploring hierarchical class activation maps for localization. *IEEE Transactions on Image Processing*, 30:5875–5888, 2021. 9

[34] S. Jo and I.-J. Yu. Puzzle-CAM: Improved localization via matching partial and full features. In *IEEE International Conference on Image Processing (ICIP)*, pages 639–643, 2021. 3, 11, 12, 22, 26, 39

[35] P. Krähenbühl and V. Koltun. Efficient inference in fully connected CRFs with gaussian edge potentials. *Advances in Neural Information Processing Systems (NeurIPS)*, 24:109–117, 2011. 10, 13

[36] V. Kulharia, S. Chandra, A. Agrawal, P. Torr, and A. Tyagi. Box2seg: Attention weighted loss and discriminative feature learning for weakly supervised segmentation. In *European Conference on Computer Vision (ECCV)*, pages 290–308. Springer, 2020. 3

[37] H. Kweon, S.-H. Yoon, H. Kim, D. Park, and K.-J. Yoon. Unlocking the potential of ordinary classifier: Class-specific adversarial erasing framework for weakly supervised semantic segmentation. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6974–6983, 2021. 3, 10, 12, 13, 22, 26, 39

[38] R. LaLonde, Z. Xu, I. Irmakci, S. Jain, and U. Bagci. Capsules for biomedical image segmentation. *Medical Image Analysis*, 68:101889, 2021. 2

[39] J. Lee, J. Choi, J. Mok, and S. Yoon. Reducing information bottleneck for weakly supervised semantic segmentation. *Advances in Neural Information Processing Systems (NeurIPS)*, 34:27408–27421, 2021. 3, 12, 26, 58

[40] J. Lee, J. Yi, C. Shin, and S. Yoon. BBAM: Bounding box attribution map for weakly supervised semantic and instance segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2643–2652, 2021. 3

[41] J. R. Lee, S. Kim, I. Park, T. Eo, and D. Hwang. Relevance-CAM: Your model already knows where to look. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14944–14953, 2021. 2, 9

[42] M. Lee, D. Kim, and H. Shim. Threshold matters in wsss: Manipulating the activation for the robust and accurate segmentation model against thresholds. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4330–4339, 2022. 26

[43] S. Lee, M. Lee, J. Lee, and H. Shim. Railroad is not a train: Saliency as pseudo-pixel supervision for weakly supervised semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5495–5505, 2021. 3, 12, 26, 58

[44] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou. A survey of convolutional neural networks: Analysis, applications, and prospects. *IEEE Transactions on Neural Networks and Learning Systems*, 33(12):6999–7019, 2022. 1, 5

[45] T. Lin, Y. Wang, X. Liu, and X. Qiu. A survey of transformers. *AI Open*, 3:111–132, 2022. 6

[46] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common Objects in Context. In *European Conference on Computer Vision (ECCV)*, pages 740–755, Cham, 2014. Springer International Publishing. 13, 19, 35

[47] J.-J. Liu, Q. Hou, M.-M. Cheng, J. Feng, and J. Jiang. A simple pooling-based design for real-time salient object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3917–3926, 2019. 15

[48] X. Liu, Y. Han, S. Bai, Y. Ge, T. Wang, X. Han, S. Li, J. You, and J. Lu. Importance-aware semantic segmentation in self-driving with discrete wasserstein training. In *AAAI Conference on Artificial Intelligence*, volume 34, pages 11629–11636, 2020. 2

[49] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, 2015. 6

[50] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, and D. Terzopoulos. Image segmentation using deep learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2021. 1, 5

[51] Y. Mo, Y. Wu, X. Yang, F. Liu, and Y. Liao. Review the state-of-the-art technologies of semantic segmentation based on deep learning. *Neurocomputing*, 493:626–646, 2022. 2

[52] R. Müller, S. Kornblith, and G. E. Hinton. When does label smoothing help? *Advances in Neural Information Processing Systems (NIPS)*, 32, 2019. 39

[53] S. J. Oh, R. Benenson, A. Khoreva, Z. Akata, M. Fritz, and B. Schiele. Exploiting saliency for object segmentation from image level labels. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5038–5047, 2017. 58

[54] Y. Oh, B. Kim, and B. Ham. Background-aware pooling and noise-aware loss for weakly-supervised semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6913–6922, 2021. 3

[55] Y. Ouassit, S. Ardchir, M. Yassine El Ghoumari, and M. Azouazi. A brief survey on weakly supervised semantic segmentation. *International Journal of Online & Biomedical Engineering*, 18(10), 2022. 2, 3, 5

[56] W. Ouyang, C. F. Winsnes, M. Hjelmare, A. J. Cesnik, L. Åkesson, H. Xu, D. P. Sullivan, S. Dai, J. Lan, P. Jinmo, S. M. Galib, C. Henkel, K. Hwang, D. Poplavskiy, B. Tunguz, R. D. Wolfinger, Y. Gu, C. Li, J. Xie, D. Buslov, S. Fironov, A. Kiselev, D. Panchenko, X. Cao, R. Wei, Y. Wu, X. Zhu, K.-L. Tseng, Z. Gao, C. Ju, X. Yi, H. Zheng, C. Kappel, and E. Lundberg. Analysis of the human protein atlas image classification competition. *Nature Methods*, 16(12):1254–1261, 2019. 20, 35

[57] J. Pons, O. Slizovskaia, R. Gong, E. Gómez, and X. Serra. Timbre analysis of music audio signals with convolutional neural networks. In *25th European Signal Processing Conference (EUSIPCO)*, pages 2744–2748. IEEE, 2017. 2

[58] W. Rawat and Z. Wang. Deep convolutional neural networks for image classification: A comprehensive review. *Neural Computation*, 29(9):2352–2449, 2017. 1

[59] S. Rossetti, D. Zappia, M. Sanzari, M. Schaerf, and F. Pirri. Max pooling with vision transformers reconciles class and shape in weakly supervised semantic segmentation. In *European Conference on Computer Vision (ECCV)*, pages 446–463. Springer, 2022. 6, 17, 26

[60] L. Ru, Y. Zhan, B. Yu, and B. Du. Learning affinity from attention: end-to-end weakly-supervised semantic segmentation with transformers. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16846–16855, 2022. 6, 17

[61] D. Samuel and G. Chechik. Distributional robustness loss for long-tail learning. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9495–9504, 2021. 5

[62] 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 *IEEE International Conference on Computer Vision (ICCV)*, pages 618–626, 2017. 2, 8, 10, 34, 42

[63] I. Shendryk, Y. Rist, R. Lucas, P. Thorburn, and C. Ticehurst. Deep learning - a new approach for multi-label scene classification in Planetscope and Sentinel-2 imagery. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 1116–1119, 2018. 19, 35

[64] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013. 7

[65] D. Smilkov, N. Thorat, B. Kim, F. B. Viégas, and M. Wattenberg. SmoothGrad: removing noise by adding noise. *arXiv*, abs/1706.03825, 2017. 7

[66] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014. 7

[67] S. Srinivas and F. Fleuret. Full-gradient representation for neural network visualization. *arXiv preprint arXiv:1905.00780*, 2019. 7

[68] Y. Su, R. Sun, G. Lin, and Q. Wu. Context decoupling augmentation for weakly supervised semantic segmentation. *arXiv*, abs/2103.01795, 2021. 49

[69] F. Sun and W. Li. Saliency guided deep network for weakly-supervised image segmentation. *Pattern Recognition Letters*, 120:62–68, 2019. 58

[70] I. Sutskever, J. Martens, G. Dahl, and G. Hinton. On the importance of initialization and momentum in deep learning. In *International Conference on Machine Learning (ICML)*, pages 1139–1147. PMLR, 2013. 36

[71] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826, 2016. 39

[72] H. Tachibana, K. Uenoyama, and S. Aihara. Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4784–4788, 2018. 2

[73] A. N. Tarekegn, M. Giacobini, and K. Michalak. A review of methods for imbalanced multi-label classification. *Pattern Recognition*, 118:107965, 2021. 16, 37

[74] G. Vilone and L. Longo. Explainable artificial intelligence: a systematic review. *arXiv preprint arXiv:2006.00093*, 2020. 8

[75] G. Vilone and L. Longo. Notions of explainability and evaluation approaches for explainable artificial intelligence. *Information Fusion*, 76:89–106, 2021. 7

[76] H. Wang, R. Naidu, J. Michael, and S. S. Kundu. SS-CAM: Smoothed Score-CAM for sharper visual feature localization. *arXiv preprint arXiv:2006.14255*, 2020. 9

[77] H. Wang, Z. Wang, M. Du, F. Yang, Z. Zhang, S. Ding, P. Mardziel, and X. Hu. Score-CAM: Score-weighted visual explanations for convolutional neural networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 24–25, 2020. 2, 9, 34

[78] T. Wang, Y. Li, B. Kang, J. Li, J. Liew, S. Tang, S. Hoi, and J. Feng. The devil is in classification: A simple framework for long-tail object detection and instance segmentation. *arXiv preprint arXiv:2007.11978*, 2020. 5

[79] Y. Wang, H. Wang, Y. Shen, J. Fei, W. Li, G. Jin, L. Wu, R. Zhao, and X. Le. Semi-supervised semantic segmentation using unreliable pseudo-labels. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4238–4247, 2022. 19, 59

[80] Y. Wang, J. Zhang, M. Kan, S. Shan, and X. Chen. Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12275–12284, 2020. 26

[81] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. Convolutional pose machines. In *IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4724–4732, 2016. 1

[82] Z. Wu, C. Shen, and A. Van Den Hengel. Wider or deeper: Revisiting the resnet model for visual recognition. *Pattern Recognition*, 90:119–133, 2019. 6, 17

[83] F. Xie, J. Yang, J. Liu, Z. Jiang, Y. Zheng, and Y. Wang. Skin lesion segmentation using high-resolution convolutional neural network. *Computer Methods and Programs in Biomedicine*, 186:105241, 2020. 2

[84] J. Xie, J. Xiang, J. Chen, X. Hou, X. Zhao, and L. Shen. C^{2}AM: Contrastive learning of class-agnostic activation map for weakly supervised object localization and semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 989–998, 2022. 14, 22, 39, 58, 62

[85] F. Xu, H. Uszkoreit, Y. Du, W. Fan, D. Zhao, and J. Zhu. Explainable AI: A brief survey on history, research areas, approaches and challenges. In J. Tang, M.-Y. Kan, D. Zhao, S. Li, and H. Zan, editors, *Natural Language Processing and Chinese Computing*, pages 563–574, Cham, 2019. Springer International Publishing. 7

[86] L. Xu, W. Ouyang, M. Bennamoun, F. Boussaid, and D. Xu. Multi-class token transformer for weakly supervised semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4300–4309, 2022. 6, 17, 26

[87] L. Yao, C. Mao, and Y. Luo. Graph convolutional networks for text classification. In *AAAI Conference on Artificial Intelligence*, volume 33, pages 7370–7377, 2019. 2

[88] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015. 6, 17

[89] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo. CutMix: Regularization strategy to train strong classifiers with localizable features. In *IEEE/CVF international conference on computer vision (CVPR)*, pages 6023–6032, 2019. 49

[90] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision (ECCV)*, pages 818–833. Springer, 2014. 2, 5

[91] Z. Zhan, X. Zhang, Y. Liu, X. Sun, C. Pang, and C. Zhao. Vegetation land use/land cover extraction from high-resolution satellite images based on adaptive context inference. *IEEE Access*, 8:21036–21051, 2020. 2

[92] B. Zhang, J. Xiao, Y. Wei, K. Huang, S. Luo, and Y. Zhao. End-to-end weakly supervised semantic segmentation with reliable region mining. *Pattern Recognition*, 128:108663, 2022. 3

[93] D. Zhang, J. Han, G. Cheng, and M.-H. Yang. Weakly supervised object localization and detection: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(9):5866–5885, 2022. 1, 3

[94] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz. MixUp: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 49

[95] H. Zhang, C. Wu, Z. Zhang, Y. Zhu, H. Lin, Z. Zhang, Y. Sun, T. He, J. Mueller, R. Manmatha, et al. ResNeSt: Split-attention networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2736–2746, 2022. 6, 12

[96] S. Zhang, Z. Li, S. Yan, X. He, and J. Sun. Distribution alignment: A unified framework for long-tail visual recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2361–2370, 2021. 5, 18

[97] B. Zhou, Q. Cui, X.-S. Wei, and Z.-M. Chen. BBN: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9719–9728, 2020. 5, 18

[98] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2921–2929, 2016. 2, 7, 10

Supplementary Materials

A Additional Details to Experimental Setup

In this section, we provide additional details over the experimental settings adopted in the experiments conducted thus far.

A.1 Explainable Artificial Intelligence

In this section, we detail the experimental procedures employed to evaluate the proposed explaining techniques with respect to the most popular alternatives found in current literature, considering multiple architectures and datasets.

A.1.1 Evaluations over Architectures and Problem Domains

In conformity with literature, we evaluate the effect of architectural change over the explanations produced from various visualization techniques by employing multiple popular alternatives of Convolutional Neural Network architectures. More specifically, we train and evaluate three architectures over Pascal VOC 2007: VGG16-GAP (VGG16), ResNet101 (RN101) and EfficientNet-B6 (EN6). We approximate the evaluation conditions of previous works [9, 62, 77] by warm-starting from weights pre-trained over the ILSVRC 2012 dataset, and fine tuning the networks over the Pascal VOC 2007 dataset [23].

We further evaluate the different visualization techniques considering five image-related problem sets, resulting in measurements and insights about the behavior and efficacy of these techniques over various scenarios. In these, it is expected that data patterns, class co-occurring groups and semantic contexts greatly differ, providing a more comprehensive understanding of these techniques. A brief summary of the employed datasets, representing the different problem sets, is provided as follows.

Pascal VOC 2007 (VOC07) The Pascal VOC 2007 dataset [23] is a well established dataset in Computer Vision and Machine Learning literature, being frequently employed in the evaluation of AI explaining methods and techniques. Comprising of 2,501 training

samples, 2,510 validation samples and 4,952 test samples, this set contains images with multiple objects belonging to 20 distinct classes.

Pascal VOC 2012 (VOC12) This dataset extends the Pascal VOC 2007 dataset to 5,717 training samples, 5,823 validation samples and 10,991 unlabeled test samples [22], while sharing the same classes with its previous version [23].

Microsoft Common Objects in Context 2017 (COCO17) The COCO 2017 dataset [46] contains 118,287 training samples, 5,000 validation samples and 40,670 unlabeled test samples. Images in this set are richly annotated with respect to various objects belonging to 80 distinct classes (classification, detection and segmentation annotations are available). Furthermore, this set respects contextual information of classes, by presenting objects in the usual environments and scenarios.

Planet: Understanding the Amazon from Space (P:UAS) This satellite imagery dataset was originally provided by Planet for a competition in the Kaggle platform, and comprises 40,479 training samples and 61,191 test samples [63]. Samples correspond to "chips" of satellite photographs of the Amazon rainforest, and are annotated with respect to their natural features (e.g., *primary forest*, *water*, *cloudy*, *haze*) one the observed human intervention in the area (e.g., *agriculture*, *road*, *selective logging*, *mining*).

Human Protein Atlas Image Classification (HPA) Firstly introduced in a Kaggle competition of same name, this set comprises 31,072 training samples and 11,702 test samples [56]. Each sample is represented by a microscopic image framing cellular bodies and proteins of interest, as well as a label set from the set of 28 available classes (e.g., Nucleoplasm, Cytosol, Plasma membrane, Nucleoli). This dataset represent many computational challenges, and it is used to measure the behavior of explaining techniques over ill-distributed datasets, recurrently found in real-case scenarios. Besides the natural difficulty of learning core visual patterns of intrinsically associated and frequently co-occurring cellular components, we observe an overwhelming class imbalanced in the training set, as well a class

distribution shift in the test set, resulting in relative low scores for all competitors in the original Kaggle challenge[6].

A.1.2 Training Procedure

Firstly, images in all datasets and experiments are resized with the preservation of their original aspect ratio, in which their shortest dimension (height or width) is matched the expected size of the visual receptive field. They are then centrally cropped along their largest dimension to the exact size of the aforementioned field (224×224 for VGG-GAP and 512×512 for ResNet101 and EfficientNetB6). In conformity with literature, we report the visualization results over the validation subset of each dataset.

Before the training procedure, weights of the convolutional pipeline are initialized with the set of weights pre-trained over ImageNet. A Global Average Pooling (GAP) layer and a *sigmoid* dense layer (with the number of units equal to the number of labels in the dataset) are then appended to the pipeline, forming the entire multi-label classification model.

Training ensues in two stages. In the first, the classification head is trained for 30 epochs with learning rate = 0.1. In the second stage, 60% of the layers of the *backbone* are unfrozen and the model is once again trained for 80 epochs using Stochastic Gradient Descent with learning rate = 0.01 and Nesterov momentum [70] equals to 0.9. Learning rate is reduced by a factor of 0.5 after every 3 epochs without decrease in validation loss. Training is halted if no improvements are observed after 20 epochs.

A.1.3 Evaluation Metrics

In order to compare our explaining techniques to current literature in a multi-label setting, we employ slightly modified versions of the metrics defined by Chattopadhay et al. [9]. Specifically, the *Increase in Confidence* (Eq. (12)) and *Average Drop* (Eq. (13)) metrics are extended to take into consideration the classification units associated with each classes present in each sample, in opposite of only considering the most intensively activating unit. Furthermore, three new metrics were designed to better evaluate the inadvertent activation

[6]Human Protein Atlas. Human Protein Atlas Image Classification. In: Kaggle. kaggle.com/competitions/human-protein-atlas-image-classification (Jan 2019). Accessed on Aug 2022.

of the produced class-specific explanation maps over objects associated with co-occurring classes. We remark that the metrics considered in this work will reduce to their conventional form, as commonly employed in literature, in single-label classification problems.

Next, we provide the formal definition of the aforementioned metrics. We remark that while the *micro-average* form was used in their respective equations for simplicity, it does not capture well the unbalanced nature of multi-label problems [73]. Hence, we report metrics in their *macro-averaged* form (or *class-frequency balanced*) in Appendix B.1.2, in which metric results are computed separately for each class and averaged, removing the impact of label frequency in the overall result.

Increase in Confidence (%IC) The rate in which masking the input image x_i by the visualization mask M_i^c has produced a higher classification score $O_{ic}^c = f(M_i^c \circ x_i)^c$ than the baseline $Y_i^c = f(x_i)^c$:

$$\frac{1}{\sum_i |C_i|} \sum_i^N \sum_{c \in C_i} [Y_i^c < O_{ic}^c] \tag{12}$$

This metric measures scenarios where removing background noise must improve classification confidence. We report results for this metric in compliance with literature, but raise the following question regarding the consistency of this metric: the classifying units of a *sigmoid* classifier are not in direct competition with each other for total activation energy, as it happens with units in *softmax* classifiers. For an ideal classifier, in which concepts are perfectly separated and no false correlation exist, one could argue that the removal of an object from an image should not affect the classification score of another object.

Average Drop (%AD) The rate of drop in the confidence of a model for a particular image x_i and label c, when only the highlighted region $M_i^c \circ x_i$ is fed to the network:

$$\frac{1}{\sum_i |C_i|} \sum_i^N \sum_{c \in C_i} \frac{\max(0, Y_i^c - O_{ic}^c)}{Y_i^c} \tag{13}$$

Average Drop expresses the idea that masking the image with an accurate mask should not decrease confidence in the label of interest, that is, it measures if your mask is correctly

positioned on top of the important regions that determine the label of interest.

Average Drop of Others (%ADO) The rate of drop in the confidence of a model for a particular image x_i and labels $n \in N_i = C_i \setminus \{c\}$, when only the highlighted region $M_i^c \circ x_i$ is fed to the network:

$$\frac{1}{\sum_i |C_i|} \sum_i^N \sum_{c \in C_i} \frac{1}{|N_i|} \sum_{n \in N_i} \frac{\max(0, Y_i^n - O_{ic}^n)}{Y_i^n} \tag{14}$$

This metric captures the effect of a mask M_i^c over objects of other labels N_i present in x_i, in which the masking of the input x_i for a given class c should cause the confidence in other labels to drop. One expects an ideal mask to not retain any objects of other classes, that is, $f(M_i^c \circ x_i)^n \approx 0, \forall n \in N_i$.

Average Retention (%AR) The rate of retention of confidence of a model for a particular image x_i and label c, when the region highlighted by the visualization map for label c is occluded:

$$\frac{1}{\sum_i |C_i|} \sum_i^N \sum_{c \in C_i} \frac{\max(0, Y_i^c - \bar{O}_{ic}^c)}{Y_i^c} \tag{15}$$

where $\bar{O}_{ic}^c = f((1 - M_i^c) \circ x_i)^c$.

While *Average Drop* measures if the map M_i^c is correctly positioned over an object of label c, *Average Retention* attempts to capture if M_i^c covers all regions occupied by objects of label c, that is, masking the input with an accurate complement mask $(1 - M_i^c)$ should decrease confidence in class c.

Average Retention of Others (%ARO) The rate of retention of confidence of a model for a particular image x_i and labels $n \in N_i$, when the region highlighted by the visualization map for label c is occluded:

$$\frac{1}{\sum_i |C_i|} \sum_i^N \sum_{c \in C_i} \frac{1}{|N_i|} \sum_{n \in N_i} \frac{\max(0, Y_i^n - \bar{O}_{ic}^n)}{Y_i^n} \tag{16}$$

This metric evaluates if the masking of input x_i for all labels but c retains the confidence

of the model in detecting these same labels. An ideal mask complement for class c should cover all objects of the other classes, that is, $f((1 - M_i^c) \circ x_i)^n \approx f(x_i)^n, \forall n \in N_i$.

F_1- and F_1+ Scores Although the considered metrics cover the various facets of the evaluation of AI explaining methods over multi-label scenarios, it may create difficulties in the analysis or interpretation of the results, requiring a high degree of attention and memorization from readers. Therefore, we opted to combine similar measurements using a harmonic mean (F_1 score). More specifically, we consider (a) F_1- as the harmonic mean between %AD and %ARO, both error measures; and (b) F_1+: the harmonic mean between %AR and %ADO, both utility functions (higher is better).

A.2 Weakly Supervised Semantic Segmentation

In accordance with literature [3,34,37], CAM-generating models are trained with Stochastic Gradient Descent (SGD) for 15 epochs with linearly decaying learning rates of 0.1 and 0.01 (for randomly initialized weights and pre-trained weights, respectively), and 1e-4 weight decay. Furthermore, samples are augmented with random resizing/cropping, while label smoothing [52,71] is applied. To improve classification robustness, we employ RandAugment (RA) [15] when training Ordinary Classifiers. Conversely, we observed a marginal decrease in mIoU when training Puzzle and OC-CSE models with RA, and thus opted to train them with simple color augmentation (variation of contrast, brightness, saturation and hue).

When training P-OC, the coefficients λ_{re} and λ_{cse} are kept as originally proposed: λ_{re} increases linearly from 0 to 4 during the first half epochs, while λ_{cse} increases linearly from 0.3 to 1 throughout training. For P-NOC, λ_{noc} increases from 0 to 1, while learning rate decreases from its initial value to 0. These settings constraint *oc* to change more significantly in the intermediate epochs, and, thus, to recognize prominent regions of objects in the first stages of training, while preventing it from learning incorrect features later on.

C^{2}AM and C^{2}AM-H are trained with the same hyper-parameters as Xie et al. [84], with the exception of the batch size, which is set to 32 for the ResNeSt269 architecture due to hardware limitations. When training C^{2}AM-H, we set δ_{fg} and λ_h to 0.4 and 1, respectively.

These values are defined after the inspection of samples of each class, confirming a low false positive rate for foreground regions in the limited inspection subset. We leave the search for their optimal values as future work.

B Detailed Discussions of Preliminary Results

In this section, we describe our preliminary results in detail. We divide them into two groups. In the first, we discuss Explainable AI methods, their performance over multi-label problems and forms to improve class-specific precision in CAMs. In the latter, we evaluate the employment of complementary strategies in WSSS problems, and derive a new method to produce more robust segmentation priors, based on the adversarial training of a strongly regularized CAM proposal network and an ordinary classifier.

B.1 Contributions for Explainable Artificial Intelligence

We list in this section our contributions towards Explainable AI.

B.1.1 Contrasting Class Gradient Information

In this section, we describe our CAM-based technique, namely MinMax-CAM, which generates visualization maps by contrasting region contributions for different classes, and thus better incorporating multi-label information into the resulting map.

Intuition Containing multiple co-occurring salient objects interacting in different contexts and obtained from various capturing conditions and settings, Multi-label problems are intrinsically more complex than the ones represented in single-label, multi-class datasets. The visual patterns associated with a given class are not necessarily the most prominent visual cue contained in their samples, while statistical artifacts, such as label co-occurrence and context, have great impact on the training and, therefore, the generalization capacity of the model. An example of such problem is remarked by Chan et al. [7]: In the extreme case in which two classes always appear together, no visual cue that effectively distinguishes

them can be learned, implying in the internalization of contextual information or correlated patterns, in opposite of the expected visual evidence for individual classes. While one can argue that the occurrence correlation of 100% between two or more classes is not a realistic scenario, fitting a classifier over frequently co-occurring classes (e.g., *dining table* and *chair* in Pascal VOC 2012 dataset [23]) might result in a significant decrease of generalization efficacy and confusing CAMs, as correlating patterns are inadvertently internalized as evidence of occurrence, thus forming false association rules.

We propose a visualization method that attempts to identify the kernel contributing regions for each label c in the input image x by averaging the signals in A^k, weighted by a combination of their direct contributions to the score of c and negative contributions to the remaining labels present in x, that is, finding regions that *maximize* the score of the label c and *minimize* the score of the remaining adjacent labels. To achieve this, we modify the gain function used by Grad-CAM to accommodate both maximizing and minimizing label groups, redefining it as the gradient of an optimization function J_c with respect to the activating signal A^k_{ij}, where J_c is the subtraction between the positive score for label c and the scores of the remaining labels represented within sample x.

Definition Let x be a sample from a dataset associated with the set of classes C_x, $c \in C_x$ a class of interest and $N_x = C_x \setminus \{c\}$. At the same time, let f be a trained convolutional network such that $A^k = [a^k_{ij}]_{H \times W}$ is the activation map for the k-th kernel in the last convolutional layer, $W = [w^c_k]$ is the weight matrix of the *sigmoid* classifying layer, containing synaptic values that linearly associate the positional signal A^k to the classification signal for class c. In these conditions, the classification score for c is given by:

$$S_c = f(x)_c = \sum_k w^c_k \frac{1}{hw}(A^k) \tag{17}$$

We consider the focused score for label c as the subtraction between the score S_c and

the average score of the remaining classes present in N_x:

$$J_c = S_c - \frac{1}{|N_x|} \sum_{n \in N_x} S_n \tag{18}$$

Finally, MinMax-Grad-CAM is defined as the combination of activation signals A^k, weighted by their respective contributions to the objective function J_c:

$$L^c_{\text{MinMax-Grad-CAM}}(f, x) = \text{ReLU}\Big(\sum_k \sum_{ij} \frac{\partial J_c}{\partial A^k_{ij}} A^k \Big) \tag{19}$$

On the other hand, we remark that J_c is a linear function with respect to $S_k, \forall k \in C_x$:

$$\frac{\partial J_c}{\partial A^k_{ij}} = \frac{\partial S_c}{\partial A^k_{ij}} - \frac{1}{|N_x|} \sum_{n \in N_x} \frac{\partial S_n}{\partial A^k_{ij}} \tag{20}$$

Hence, MinMax-Grad-CAM can be rewritten in its more efficient and direct "CAM form" (as demonstrated by Selvaraju et al. [62]), for convolutional networks where the last layer is a linear classifier. In this form, Equation (19) simplifies to:

$$L^c_{\text{MinMax-CAM}} = \text{ReLU}\Big(\sum_k \big[w^c_k - \frac{1}{|N_x|} \sum_{n \in N_x} w^n_k \big] A^k \Big) \tag{21}$$

In conformity with the literature, we employ the ReLU function in both forms, only retaining regions that positively contribute to the maximization of function J_c.

Reducing Noise by Removing Negative Contributions Let $g^k = \text{GAP}(A^k_{ij})$ be a positional-invariant signal describing the evidence of occurrence for a given data pattern k. If the ReLU activation function (or any other non-negative function) is used in the last convolutional layer, then g^k is positive, and $w^c_k > 0$ invariably associate the classification of class c to kernels that positively contribute to it. Conversely, $w^c_k < 0$ indicate kernels that negatively contribute to the classification of c.

When the contributions for classes $n \in N_x$ are naively subtracted in Equations (18) and (21), negative weights (or gradients) become positive, producing inadvertently a resid-

ual highlighting over regions that negatively contribute for the classification of n. We can mitigate this noise by decomposing the contribution factors a_k^c into (a) *positive*, that positively contribute for the classification of c, (b) *negative*, that positively contribute for the classification of $n \in N_x$, and (c) *overall negative*, that negatively contribute for the classification of all classes, frequently overlapping *background* regions in our experiments.

An alternative form (which we denote as D-MinMax-Grad-CAM, for the remaining of this work) can then be formally defined as:

$$L^c_{\text{D-MinMax-Grad-CAM}} = \text{ReLU}\Big(\sum_k \alpha_k^c A^k\Big) \tag{22}$$

where

$$\alpha_k^c = \sum_{ij}\left[\text{ReLU}\Big(\frac{\partial S_c}{\partial A_{ij}^k}\Big) - \frac{1}{|N_x|}\text{ReLU}\Big(\sum_{n\in N_x}\frac{\partial S_n}{\partial A_{ij}^k}\Big) + \frac{1}{|C_x|}\min\Big(0, \sum_{n\in C_x}\frac{\partial S_n}{\partial A_{ij}^k}\Big)\right] \tag{23}$$

Finally, a CAM derivation is also possible:

$$\alpha_k^c = \Big[\text{ReLU}(w_k^c) - \frac{1}{|N_x|}\text{ReLU}(\sum_{n\in N_x} w_k^n) + \frac{1}{|C_x|}\min(0, \sum_{n\in C_x} w_k^n)\Big] \tag{24}$$

Figures 5 and 6 exemplify visualization maps obtained from the application of various techniques over a few samples in the Pascal VOC 2012 and VOC 2007 datasets, respectively. While Grad-CAM++ and Score-CAM generated confusing maps, in which the explaining signal overflow the boundaries of the object of interest and even cover large portions of the scenario, MinMax-CAM produced more focused activation maps, where class-specific highlighting avoided objects of different classes. Meanwhile, D-MinMax-CAM has effectively reduced the residual activation over non-salient objects and background regions.

B.1.2 Quantitative Results

In this section, we report the evaluation results for well-established CAM-based techniques (CAM, Grad-CAM++, Score-CAM), while comparing them to MinMax-CAM and D-MinMax-CAM. We then discuss the properties and limitations of our technique.

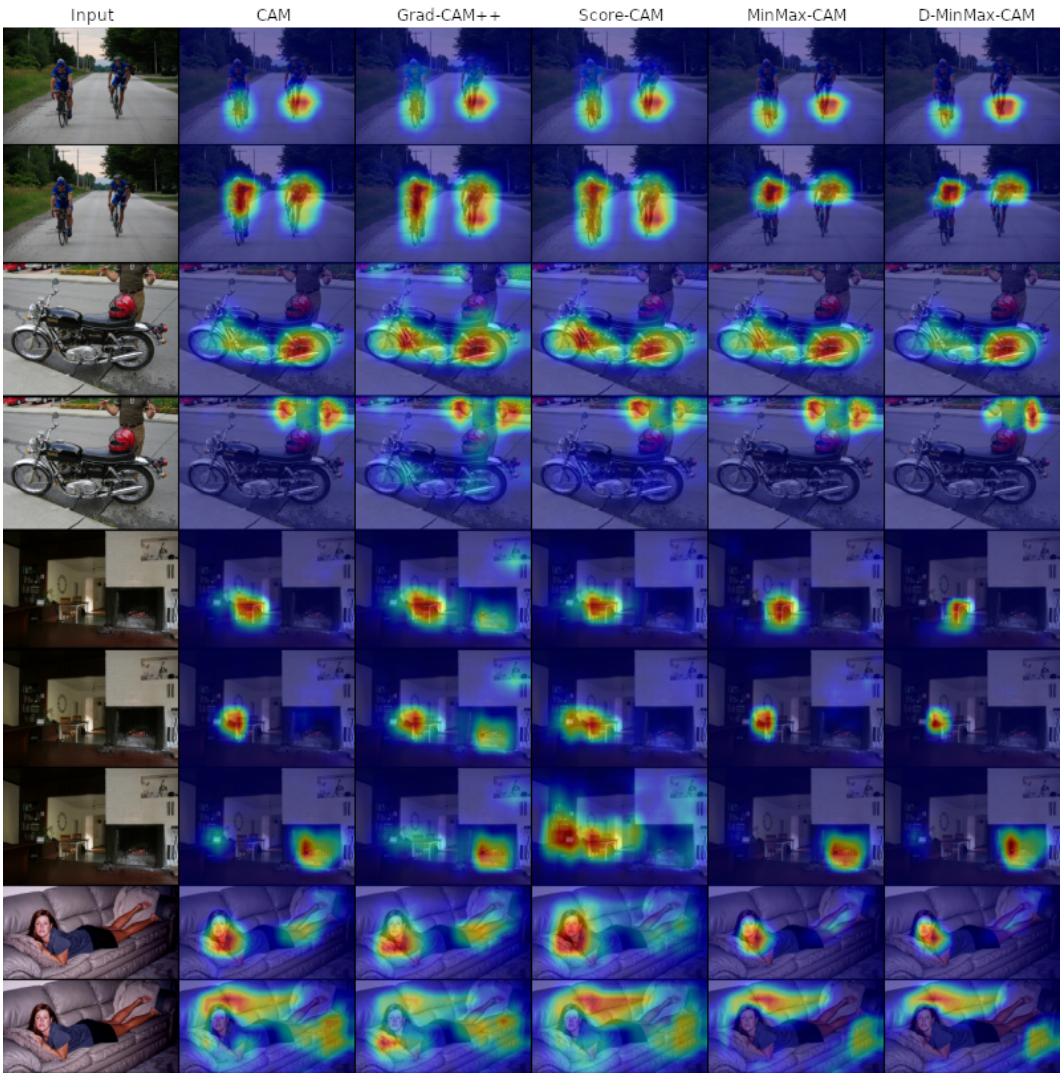


Figure 5: CAMs produced by various CAM-based methods over Pascal VOC 2012 dataset. Predictions are, from top to bottom: *bicycle*, *person*, *motorbike*, *person*, *dining table*, *chair*, *tv*, *person* and *sofa*. Source: David et al. [16].

Evaluation Over Distinct Architectures Table 3 enumerates these results over VOC07 validation set, considering the EfficientNet-B6 (Eb6), ResNet-101 (RN101) and VGG16-GAP (VGG16) architectures. We observe that Grad-CAM++ and Score-CAM result in the highest %IC for most architectures (two out of three). For EfficientNet-B6, CAM obtained the highest value for this metric (39.67%), closely followed by D-MinMax-CAM (39.49%).

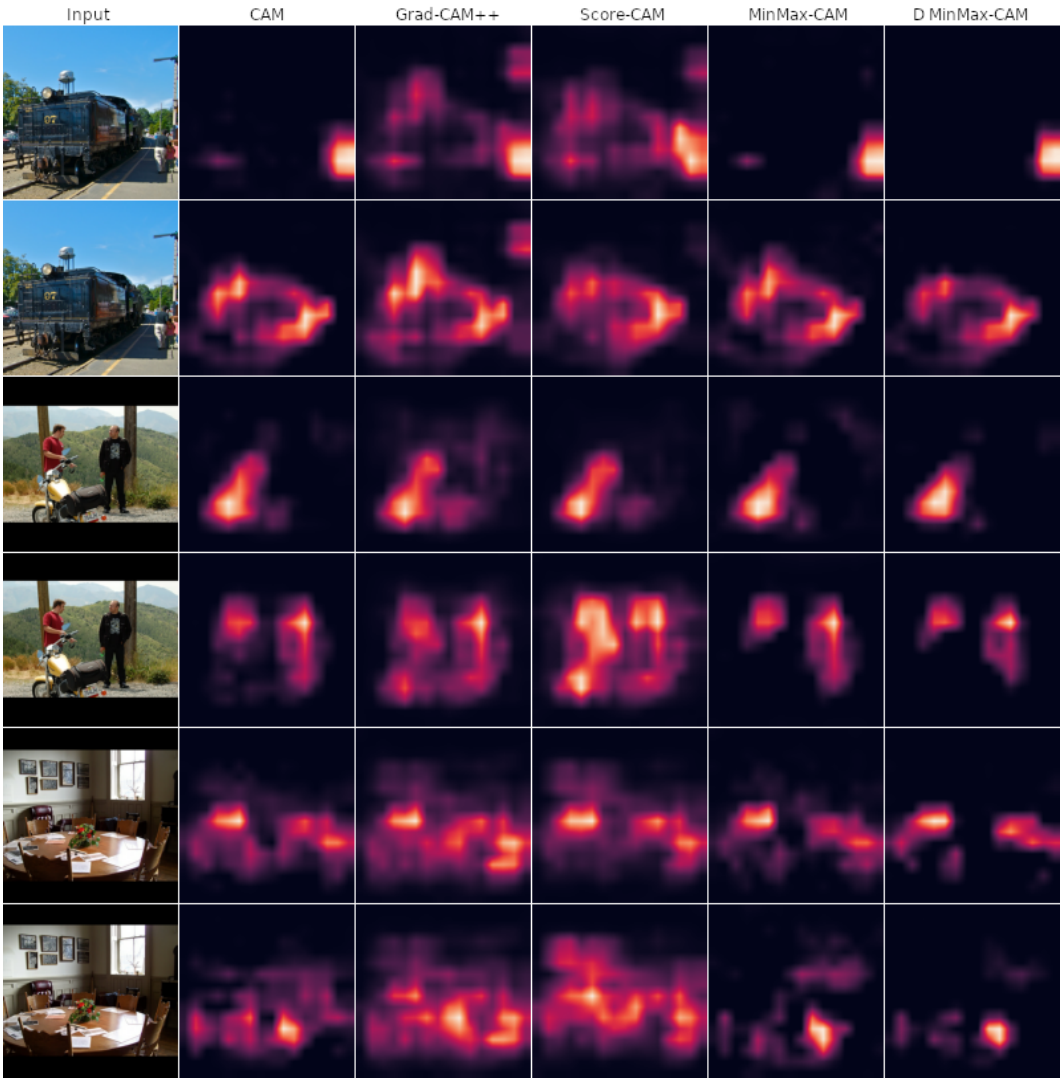


Figure 6: Attention maps produced by various CAM-based methods over Pascal VOC 2007 dataset. Predictions are, from top to bottom: *person*, *train*, *motorbike*, *person*, *chair*, and *dining table*. Source: David et al. [16].

For the remaining architectures, MinMax-CAM and D-MinMax-CAM present slightly lower %IC than CAM, while always losing to Grad-CAM++ and Score-CAM.

CAM, Grad-CAM++ and Score-CAM obtain the best %AD and %AR scores, as these metrics favor methods producing diffuse activation maps. Grad-CAM++ and Score-CAM obtained a significantly lower %AD compared to the remaining techniques, while CAM

Table 3: Score report for multiple architectures over the VOC07 dataset. Source: David et al. [16]

Metric	Model	CAM	Grad-CAM++	Score-CAM	MinMax-CAM	D-MinMax-CAM
%IC	Eb6	**39.67%**	25.13%	30.50%	34.23%	39.49%
	RN101	27.68%	31.03%	**40.76%**	26.61%	23.83%
	VGG16	5.65%	8.27%	**12.78%**	4.18%	3.76%
%AD	Eb6	22.94%	36.87%	**22.10%**	28.09%	23.71%
	RN101	25.24%	17.90%	**10.79%**	32.58%	39.25%
	VGG16	39.34%	29.22%	**19.27%**	46.78%	50.34%
%ADO	Eb6	29.43%	19.35%	20.17%	**39.82%**	31.99%
	RN101	32.73%	12.48%	14.72%	44.03%	**46.49%**
	VGG16	29.61%	18.52%	15.74%	39.33%	**39.50%**
%AR	Eb6	**11.74%**	8.40%	9.92%	10.50%	9.10%
	RN101	**16.54%**	14.04%	14.94%	14.27%	12.00%
	VGG16	40.38%	39.04%	**42.70%**	33.82%	31.00%
%ARO	Eb6	1.61%	2.53%	2.28%	**0.99%**	1.47%
	RN101	2.44%	3.94%	3.43%	1.28%	**1.16%**
	VGG16	8.84%	12.10%	12.96%	3.47%	**3.34%**
F_1-	Eb6	2.82%	4.54%	1.91%	**1.86%**	2.64%
	RN101	4.05%	5.62%	**2.20%**	2.38%	2.21%
	VGG16	13.52%	15.39%	13.42%	6.23%	**6.00%**
F_1+	Eb6	**15.79%**	10.14%	5.96%	15.40%	12.96%
	RN101	**20.84%**	11.97%	6.89%	19.85%	17.13%
	VGG16	31.70%	23.50%	22.19%	**32.16%**	29.94%

obtained marginally higher %AR scores than both MinMax alternatives, indicating that Grad-CAM++ and Score-CAM are better at covering the characteristic sections of objects, while CAM and MinMax produce activation maps with lower relative coverage.

Conversely, MinMax consistently achieves better results for %ADO and %ARO, as these metrics favor methods that produce more focused class-specific maps. When considering the F_1- metric, MinMax result in the best scores for two out of the three architecture, scoring significantly lower than CAM and Grad-CAM++, which further indicates that they are quite successful at removing regions containing objects associated to the classes N_x, while still focusing on determinant regions for the classification of c. Finally, while Score-CAM presents the best F_1- score for the RN101 architecture (2.20%), MinMax and D-MinMax-CAM closely approximate this result (2.38% and 2.21%, respectively).

CAM and MinMax-CAM present the highest F_1+ score, closely followed by D-MinMax-CAM. Moreover, the Grad-CAM++ and Score-CAM techniques present noticeably lower

scores for this metric, indicating that CAM, MinMax-CAM and D-MinMax-CAM are more successful in covering large portions of objects associated with class c without spreading over objects of adjacent classes.

Evaluation Over Distinct Problem Domains Table 4 displays results for the various explaining methods and datasets. Once again, CAM, Grad-CAM++ and Score-CAM produce the best %IC, %AD and %AR values. We attribute this to the proclivity of these techniques to retain large portions of the image, maintaining contextual information of the sample. Conversely, D-MinMax-CAM wins against the literature techniques by a large margin when considering %ADO, %ARO and F_1- score. Finally, CAM and MinMax-CAM present similar results for F_1+ score, consistently ahead of Grad-CAM++ and Score-CAM.

CAM, Grad-CAM++, MinMax-CAM and D-MinMax-CAM were evaluated in under 30 minutes, when considering the Pascal VOC 2007, VOC 2012 and MS COCO 2017 datasets, with no significant difference in performance being observed between them. Conversely, Score-CAM entailed a considerable higher execution time, considering its high computational footprint, taking approximately 16 hours and 29 hours to complete over VOC07 and P:UAS, respectively, and over 59 hours to complete over COCO17 and HPA.

B.1.3 Reducing Shared Information between Classifiers

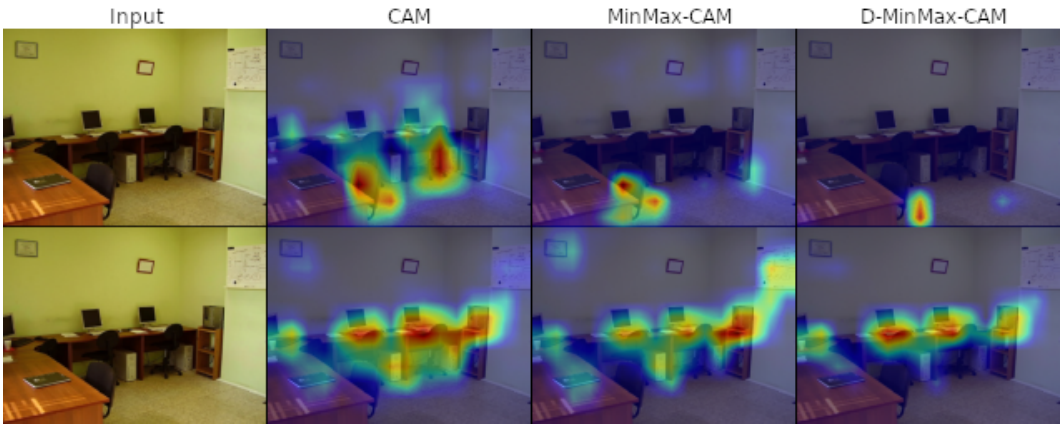
MinMax-CAM works under the assumption that two distinct classes are not associated with the same set of visual cues present in a single region in the input image. Hence, the contributions being subtracted are associated with different parts of the spatial signal A^k, and the resulting map is more focused than its counterpart generated by CAM. This assumption does not hold when a network has not learned sufficiently discriminative patterns for both labels, which can be caused by an unbalanced set or a subset of frequently co-occurring labels [7]. For instance, *tvs* frequently co-occur with *chairs*, which may induce the model to correlate the occurrence of the latter with the classification of a former, hence degenerating CAMs (Figure 7a).

Although class co-occurrence and contextual information might present useful informa-

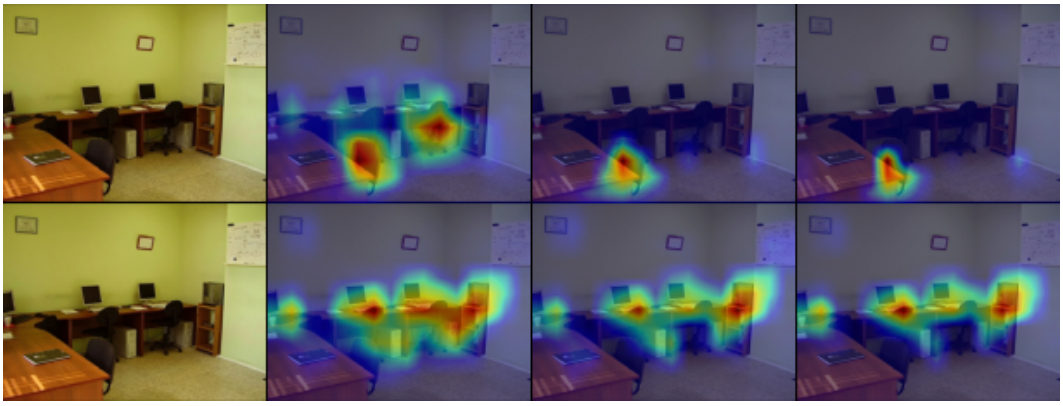
Table 4: Report of metric scores over multiple datasets. Results expanded from David et al. [16]

Metric	Dataset	CAM	Grad-CAM++	Score-CAM	MinMax-CAM	D-MinMax-CAM
%IC	P:UAS	6.09%	7.05%	**11.59%**	6.22%	6.27%
	COCO17	30.21%	32.98%	**44.69%**	23.12%	19.20%
	VOC07	27.68%	31.03%	**40.76%**	26.61%	23.83%
	VOC12	27.75%	25.40%	**35.10%**	24.70%	21.66%
	HPA	8.64%	9.29%	**11.27%**	7.63%	5.89%
%AD	P:UAS	55.25%	49.00%	**43.37%**	64.24%	66.88%
	COCO17	27.42%	17.56%	**9.62%**	40.22%	47.43%
	VOC07	25.24%	17.90%	**10.79%**	32.58%	39.25%
	VOC12	24.47%	18.69%	**10.60%**	29.17%	34.22%
	HPA	49.78%	47.02%	**41.50%**	54.16%	64.21%
%ADO	P:UAS	43.61%	33.67%	34.06%	60.04%	**60.62%**
	COCO17	51.49%	20.59%	24.45%	68.04%	**71.90%**
	VOC07	32.73%	12.48%	14.72%	44.03%	**46.49%**
	VOC12	36.44%	14.92%	18.46%	43.65%	**45.02%**
	HPA	24.01%	18.95%	17.07%	29.46%	**39.50%**
%AR	P:UAS	46.42%	**49.45%**	48.01%	37.16%	32.74%
	COCO17	**27.70%**	25.60%	26.64%	24.44%	22.79%
	VOC07	**16.54%**	14.04%	14.94%	14.27%	12.00%
	VOC12	**16.23%**	14.71%	16.22%	14.60%	13.06%
	HPA	29.15%	28.49%	**30.59%**	25.60%	15.44%
%ARO	P:UAS	25.48%	29.46%	28.13%	20.84%	**18.55%**
	COCO17	5.26%	7.92%	7.71%	3.31%	**3.13%**
	VOC07	2.44%	3.94%	3.43%	1.28%	**1.16%**
	VOC12	2.29%	3.76%	3.32%	1.21%	**1.14%**
	HPA	6.69%	9.32%	10.56%	3.60%	**1.32%**
F_1-	P:UAS	30.68%	32.07%	28.46%	28.35%	**26.42%**
	COCO17	8.23%	9.94%	7.39%	5.82%	**5.64%**
	VOC07	4.05%	5.62%	**2.20%**	2.38%	2.21%
	VOC12	3.89%	5.70%	4.30%	2.26%	**2.17%**
	HPA	10.89%	14.26%	15.10%	6.45%	**2.54%**
F_1+	P:UAS	39.54%	35.11%	35.41%	**41.00%**	37.01%
	COCO17	34.05%	21.45%	23.82%	**34.07%**	32.44%
	VOC07	**20.84%**	11.97%	6.89%	19.85%	17.13%
	VOC12	**21.25%**	13.87%	16.39%	20.25%	18.60%
	HPA	**22.85%**	18.30%	18.29%	22.71%	18.79%

tion towards the improvement of classification efficacy, these artifacts tend to cause unexpected highlighting in regions that do not contain the objects associated with classes of interest. Hence, they may also imply in the diminishing the precision of localization cues provided by CAMs, increasing the number of false positive pixels [7].



(a)



(b)

Figure 7: (a) Failure example in VOC07: contributing regions for *chair* collide with the ones for *tv*. (b) CAMs from a model trained with KUR. Source: David et al. [16].

Imprecise CAMs are mitigated by solutions that reinforce the learning of patterns that exclusively describe one or few classes, while penalizing the internalization of contextual patterns, which describe more than a single class at the same time. Examples are the various augmentation strategies based on sample combination, such as MixUp [94] and CutMix [89]); the context decoupling strategy proposed by Su et al. [68], in which objects are pasted outside their usual context; and the experiments conducted by Chan et al. [7], which evaluated the effect of "balancing" the class distribution — by removing samples containing highly correlating labels — over the DeepGlobe segmentation task [17].

Conversely to the aforementioned data-based strategies, we propose an architectural

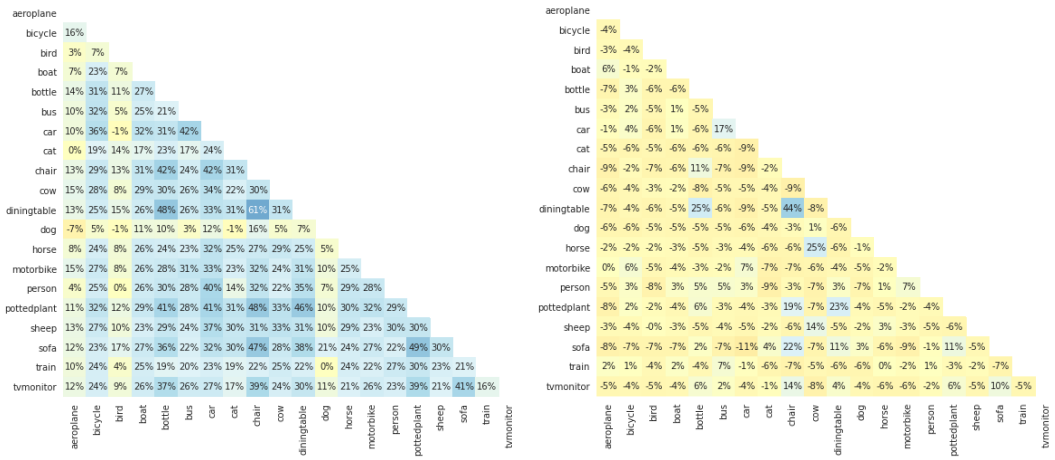
change that reinforces positive and sparse values in the weight matrix W, while striving for mutually exclusive usage of the visual signals g^k. These properties are simple and intuitive: The occurrence of visual evidence associated with classes in $C_x \setminus \{c\}$ should not affect the classification score of a given class c. At the same time, invariance between classification score and the absence of evidence of other classes can be reinforced by discouraging the formation of negative associations (weights).

Let K be the number of kernels in the last convolutional layer, C be the number of classes in the dataset, $g = [g^k]_K$ be the feature vector obtained from the pooling of last convolutional layer, $W = [w_k^c]_{K \times C}$ and $b = [b_c]_C$ the weights from the last dense layer and σ the *sigmoid* function. We define the regularization of the weights of the *sigmoid* classifier, namely Kernel Usage Regularization (KUR), as follows:

$$\begin{aligned} W^r &= W \circ \text{softmax}(W) \\ y &= \sigma(g \cdot W^r + b) \end{aligned} \tag{25}$$

When *softmax* is applied over each vector W_k, high values w_k^c — implying a strong association between g^k and S_c — will induce $softmax(w_k)^c \approx 1$, and thus $w_k^{c^r} \approx w_k^c$. As the *softmax* function quickly saturates over a few large values, the remaining associations quickly tend to 0, erasing the influence of the activation signals A^k over $S_n, \forall n \in [0, C] \setminus c)$. Finally, negative values w_k^c should have low $softmax(w_k)^c$, hence $w_i^{c^r} \approx 0$.

Figure 7b illustrates CAMs learned by a model trained with KUR. As the simultaneous usage of same kernels for distinct classification units have been regularized, subtracting contributions no longer distort the maps for any of the labels. Activations for the class *chair*, in special, are no longer shifted onto the floor. Moreover, Figure 8 illustrates the correlation between the weight classifying vectors, for both *vanilla* and KUR models. Classifying vectors are much less correlated for the model trained with KUR, indicating they are now effectively using distinct activation signals in their decision process.



(a) (b)

Figure 8: Correlation between weight vectors of *sigmoid* classifying layers of (a) an unregularized model, and (b) a model whose training was regularized with KUR.

B.1.4 Counterbalancing Activation Vanishing

In spite of the observed effectiveness in separating the available kernels between the classifying units for the VOC 2007, VOC 2012 and P:UAS datasets, its decrease in F1 score over the COCO 2017 dataset is troublesome. Upon closer inspection of this particular model, we observed that *kernel usage regularization* inadvertently causes the weights to vanish when the number of classifying units is high. This is due to the *softmax* function being initially evenly-distributed, with $\text{softmax}_c(x) \approx \frac{1}{c}$. Hence, for a large number of classes c, the initial weights are aggressively pushed towards zero, which obstructs the training process and severely compromises the solution candidate found.

However, we can counter-balancing the effect of the initial configuration of the *softmax* function over the signal distribution by simply multiplying the regularized weights by a scaling factor α, resulting in the restoration of signal's variance. For $\alpha = C$ (the number of classes), we expect weights to sustain their original variance, as $c \times \text{softmax}_c(w) \approx c\frac{1}{c} = 1$. Figure 9 illustrates the weight distribution for the baseline, KUR and KUR-α, for $\alpha = C$.

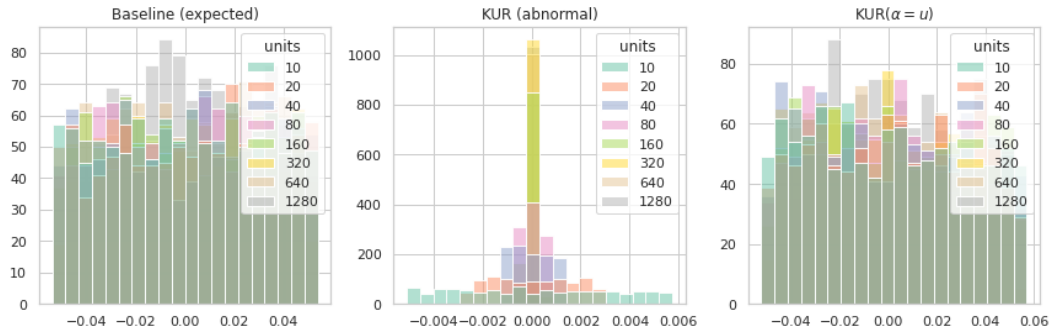


Figure 9: Weight distribution for vanilla and (kernel usage) regularized weights, for multiple output units.

Table 5: Multi-label classification score over multiple datasets, considering the baseline and regularized (KUR) models. Results expanded from David et al. [16]

Metric	Dataset	Baseline	KUR
F_1	VOC07 Test	84.26%	**85.85%**
F_1	VOC12 Val	85.05%	**85.90%**
F_2	P:UAS Val	87.80%	**88.24%**
F_2	P:UAS Private Test	89.22%	**89.81%**
F_2	P:UAS Public Test	89.62%	**90.10%**
F_1	COCO17 Val	**75.64%**	74.23%
F_1	HPA Private Test	**36.05%**	35.54%
F_1	HPA Public Test	**39.72%**	39.46%

B.1.5 Results

Table 5 reports the F_1 and F_2 scores over validation and test sets (when available) for both baseline and regularized models. We see a slight increase in F_1 and F_2 score in most cases, indicating that this regularization has positive impact on overall score of the classifier. Conversely, a noticeable decrease in score can be observed for the COCO17 dataset, which is associated with the high number of classes present in this set, implying an aggressively regularized training. By retraining the RN101 architecture over the COCO17 dataset, regularized with KUR-α s.t. $\alpha = 80$, we obtain a F_1 score of 75.55%. Finally, a decrease in F_1 score when evaluated the vanilla and KUR models over the HPA private and public test subsets is also noticeable, although small. We hypothesize that better results can be achieved with a careful finetune of hyperparameters (such as *learning rate* and α).

Examples of Class-specific Activation Maps extracted from COCO17 dataset by various

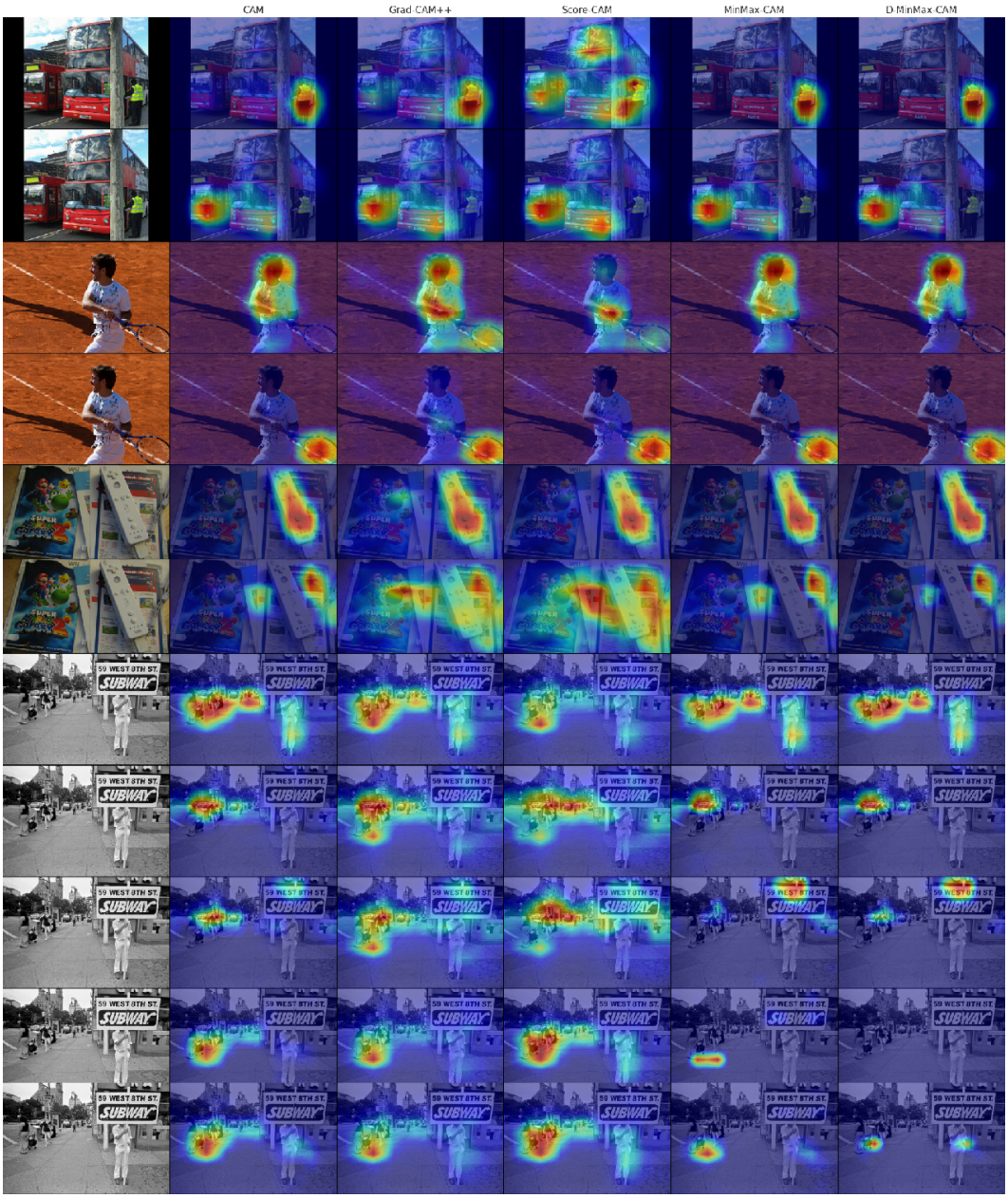


Figure 10: CAMs generated for the MS COCO dataset.

visualization techniques are illustrated in Figure 10. Once again, we observe more focused visualization maps for MinMax-CAM and D-MinMax-CAM: the *persons* next to the *buses* (first two rows) and the *tennis racket* (third and forth rows), as well as the multiple objects

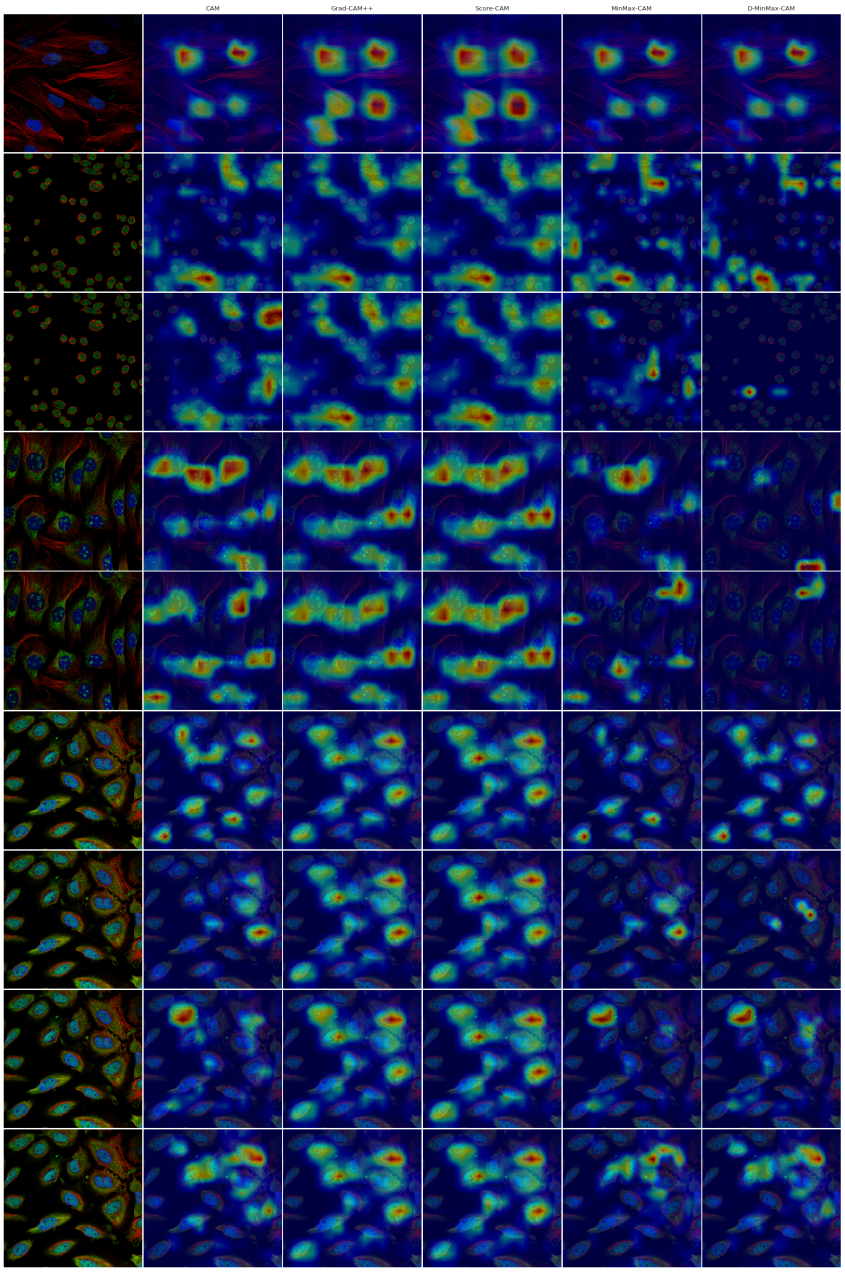


Figure 11: CAMs generated for the HPA dataset.

in the street scenario (last four rows). On the other hand, examples of visualization maps extracted from the HPA dataset are presented in Figure 11. We observe Grad-CAM++, Score-CAM, and CAM, but to a lower extent, producing similar explaining maps for many

of the examples of this set, which is also supported by their close score results reported in Table 4. Maps for different classes in the same sample seem to frequently highlight the same salient regions, indicating these are, indeed, not class-specific. At the same time, MinMax-CAM presents distinct class-specific maps in a single sample, for most examples.

Table 6 displays the results over multiple datasets, employing a RN101 network trained with KUR. Score-CAM score the highest for %IC on all but one set, while Grad-CAM++ obtains the second place among most evaluations. CAM closely follows the two best-placed techniques, while achieving the best score over the P:UAS dataset. D-MinMax-CAM shows the best F_1- scores in all datasets but one, staying in third place with a difference of 0.53 percent points from the winner (Score-CAM). Finally, MinMax-CAM and D-MinMax-CAM showed the best results in 3 out of 5 tests for the F_1+ score, while achieving a similar score to the winner (CAM) over VOC07, and the worst results when evaluated over HPA.

When comparing the results from regularized models to the ones obtained from their unregularized counterparts, we observe an overall increase in both %IC and F_1+ score for most CAM techniques and datasets. Notwithstanding, F_1- score results improved for 9 out of 25 tests, while staying relatively similar over VOC07 and VOC12. Finally, it is noticeable the decrease in difference between the results from MinMax-CAM and D-MinMax-CAM, across all metrics and datasets. This can be attributed to the regularization factor, which penalizes the existence of negative weights, approximating $\max(0, w_k^c)$ to w_k^c and, thus, D-MinMax-CAM to MinMax-CAM.

B.2 Contributions on Weakly Supervised Segmentation

In this section, we discuss forms to combine and extend WSSS methods. In order to provide a fairer comparison with literature and more reliable estimation of the effectiveness of the approach over truly WSSS problems, we re-evaluate Puzzle without the aforementioned early stopping mechanism in Table 7. For the remaining of this work, we denote this "fair" alternative as Puzzle$^{\text{f}}$ (or P$^{\text{f}}$).

Table 6: Report of metric scores per visualization technique, over multiple datasets. Classification models were regularized with KUR during training. Results expanded from David et al. [16]

Metric	Dataset	CAM	Grad-CAM++	Score-CAM	MinMax-CAM	D-MinMax-CAM
%IC	P:UAS	**15.60%**	14.39%	14.13%	11.43%	11.54%
	COCO17	34.43%	36.81%	**37.87%**	21.47%	21.49%
	VOC07	28.71%	28.07%	**34.93%**	23.90%	24.99%
	VOC12	33.32%	34.90%	**37.30%**	29.54%	29.36%
	HPA	11.19%	15.73%	**17.55%**	10.31%	5.79%
%AD	P:UAS	42.51%	42.67%	**39.50%**	51.96%	52.53%
	COCO17	22.52%	19.86%	**13.91%**	41.29%	41.39%
	VOC07	22.89%	18.65%	**11.69%**	29.80%	34.19%
	VOC12	16.09%	15.32%	**10.46%**	22.22%	22.85%
	HPA	46.41%	42.61%	**39.81%**	49.92%	59.99%
%ADO	P:UAS	38.34%	35.46%	35.21%	**49.58%**	49.51%
	COCO17	46.97%	37.63%	25.57%	69.17%	**69.28%**
	VOC07	37.30%	20.06%	17.27%	47.16%	**48.60%**
	VOC12	29.66%	21.89%	15.95%	42.07%	**42.46%**
	HPA	27.23%	21.51%	20.76%	32.38%	**33.43%**
%AR	P:UAS	**47.28%**	46.50%	43.61%	43.17%	43.01%
	COCO17	**34.40%**	34.21%	28.13%	30.05%	30.04%
	VOC07	**18.64%**	17.35%	16.91%	16.02%	14.72%
	VOC12	**18.66%**	18.37%	17.72%	17.10%	16.99%
	HPA	26.49%	**26.52%**	25.73%	23.91%	13.57%
%ARO	P:UAS	25.43%	26.35%	26.80%	20.79%	**20.72%**
	COCO17	7.14%	7.85%	11.36%	4.24%	**4.23%**
	VOC07	2.44%	3.45%	3.95%	1.35%	**1.22%**
	VOC12	2.59%	2.89%	4.00%	1.22%	**1.20%**
	HPA	7.62%	10.12%	10.24%	5.00%	**1.53%**
F_1-	P:UAS	27.02%	27.68%	**26.62%**	26.86%	27.15%
	COCO17	10.08%	10.38%	11.15%	**7.33%**	**7.33%**
	VOC07	4.12%	5.41%	2.69%	2.47%	**2.28%**
	VOC12	3.97%	4.30%	4.96%	2.24%	**2.21%**
	HPA	11.86%	14.03%	13.48%	8.49%	**2.92%**
F_1+	P:UAS	36.53%	35.05%	34.46%	**39.15%**	39.03%
	COCO17	38.08%	34.42%	25.19%	40.64%	**40.65%**
	VOC07	**23.89%**	17.87%	8.10%	22.38%	20.97%
	VOC12	21.99%	19.28%	16.24%	**22.84%**	22.78%
	HPA	**23.52%**	20.56%	19.96%	23.38%	15.65%

B.2.1 Combining Regularizing Strategies

While Puzzle expands the spatial activation signal onto all parts of salient objects, OC-CSE regularizes the contours and boundaries of the produced activations, resulting in a better separation between objects of different classes. Thus, we remark these two tech-

niques as complementary, and we raise the hypothesis that combining them can mitigate the class-specificity problem found in Puzzle, while maintaining a high *completeness* and, thus, implying on more precise semantic segmentation proposals. Formally, we define the P-OC training strategy as the optimization of the following objective functions:

$$\begin{aligned}\mathcal{L}_{\text{P-OC}} &= \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{re-cls}} + \mathcal{L}_{\text{re}} + \lambda_{\text{cse}}\mathcal{L}_{\text{cse}} \\ &= \ell_{\text{bce}}(p_i, y_i) + \ell_{\text{bce}}(p_i^{\text{re}}, y_i) + \lambda_{\text{re}}\|A_i - A_i^{\text{re}}\|_1 + \lambda_{\text{cse}}\ell_{\text{bce}}(\hat{p}_i, \hat{y}_i)\end{aligned} \tag{26}$$

where y_i and p_i are the target and estimated posterior probabilities associated with sample x_i, respectively; and $\hat{p}_i$ is the posterior probability vector predicted by the *oc*, when presented with the x_i masked by the activation mask of c_k, and $\hat{y}_i = y_i \setminus \{c_k\}$.

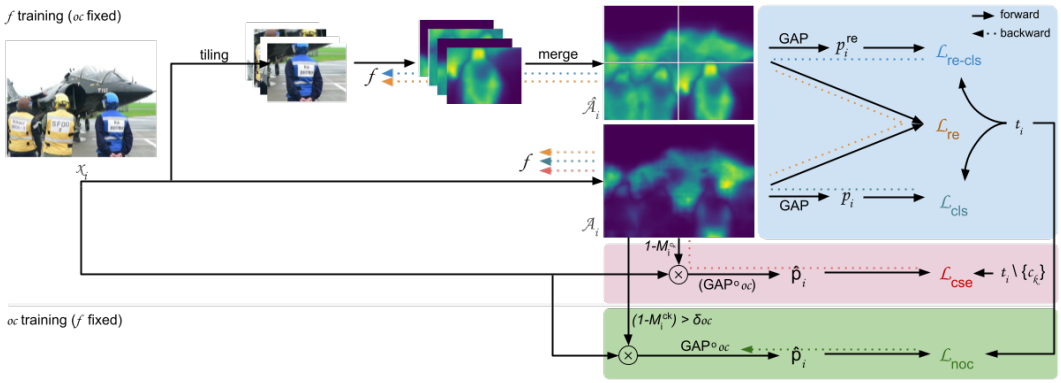


Figure 12: Overview of our adversarial training setup, in which f is optimized considering both Puzzle module and the ordinary classifier *oc*. f is sub-sequentially fixed and *oc* is updated to shift its attention towards regions currently ignored by f.

Subsequently, we consider an adversarial training setup where *oc* is gradually fine-tuned to associate images containing partially masked objects to their original classes, making *oc* a "not so ordinary" classifier. This strategy (namely P-NOC), illustrated in Figure 12 and detailed in Algorithm 1.

In summary, P-NOC is trained by alternatively optimizing two objectives:

$$\mathcal{L}_f = \mathbb{E}_{(x,y)\sim\mathcal{D},r\sim y}[\mathcal{L}_{\text{P}} + \lambda_{\text{cse}}\ell_{\text{cls}}(p^{\text{oc}}, y \setminus \{r\})] \tag{27}$$

$$\mathcal{L}_{\text{noc}} = \mathbb{E}_{(x,y)\sim\mathcal{D},r\sim y}[\lambda_{\text{noc}}\ell_{\text{cls}}(p^{\text{noc}}, y)] \tag{28}$$

where $p^{\text{noc}} = oc(x \circ (1 - \psi(A^r) > \delta_{\text{noc}}))$.

By refining *noc* to match the masked image to the label vector y, in which $y^r = 1$, we expect it to gradually shift its attention towards secondary (and yet discriminative) regions, and, thus, to provide more useful regularization to the training of the generator. Concomitantly, we expect f to not forget the class discriminative regions learned so far, considering (a) its learning rate is linearly decaying towards 0; and (b) the degeneration of the masks would result in an increase of $\mathcal{L}_{\text{cse}}$.

Algorithm 1 Proposed P-NOC algorithm

Require: Training set $\mathcal{D} = \{\mathcal{X}, \mathcal{Y}\}$, CAM generating networks f and *noc*, $k_{noc} \in \mathbb{N}$, $\delta_{\text{noc}} \in (0, 1)$

$i \leftarrow 0$
while not done **do**
  Sample a batch (x_i, y_i) from $\mathcal{D}$, and r from y_i
  // Fix noc and train f
  Compute $A_i^c = f(x_i), A^{\text{re}c}_i = \text{merge}(f(\text{tile}(x_i)))$
  Compute $\mathcal{L}_{\text{P-OC}}$ loss from Eq. (26)
  Update weights of f by $\nabla\mathcal{L}_{\text{P-OC}}$
  $i \leftarrow i + 1$
  if $i \mod k_{noc} = 0$ **then**
    // Fix f and train noc
    $\hat{x}_i = x_i \circ (1 - \psi(A_i^r) > \delta_{\text{noc}})$
    Compute $\mathcal{L}_{noc}$ from Eq. (27)
    Update weights of *noc* by $\nabla\mathcal{L}_{\text{noc}}$
  end if
end while

B.2.2 Deriving Saliency Information

The considerable improvements obtained by C^{2}AM [84], as well as various works in related literature [10, 32, 39, 43, 53, 69], indicate that the utilization of saliency information as complementary information is advantageous for the solution of WSSS tasks. We are thus encouraged to expand upon C^{2}AM.

We propose to utilize *saliency hints* extracted from models trained in the weakly supervised scheme. Our approach is inspired by recently obtained results in the task of Semi-Supervised Semantic Segmentation, in which a teacher network is used to provide additional

annotation for the training of a student network [79]. More specifically, we leverage the segmentation prior generating models previously trained to extract the spatial Class-Specific Activation Maps A_i^k (CAMs), for every image x_i in the dataset. The maps are interpolated to match the spatial sizes of x_i, and sub-sequentially reduced, where the *max pooling* operation is applied onto k, resulting in maps that hint (most likely) salient regions. Given the previously observed lack of *completeness* in CAMs, only regions associated with a high activation intensity are considered as *fg* hints, and thus used to reinforce a strong output classification value for the disentangling branch.

We define C^{2}AM-H as an extension of C^{2}AM, in which *fg* hints are employed to guide training towards a solution in which salient regions are associated with high prediction values from d (anchored), and all salient objects are contained within the same partition. In practice, this implies in the addition of a new objective function in Eq. (7): the cross-entropy loss term between the collected hints $\hat{y}_i$, for $i \in [b, b+n)$ and the posterior probability predicted by d. C^{2}AM-H is trained with the following loss function:

$$\mathcal{L}^{\mathcal{B}}_{\text{C}^2\text{AM-H}} = \mathcal{L}^{\mathcal{B}}_{\text{pos-f}} + \mathcal{L}^{\mathcal{B}}_{\text{pos-b}} + \mathcal{L}^{\mathcal{B}}_{\text{neg}} + \lambda_h \sum_{i \in b} \sum_{h,w} \mathbb{1}_{[A_i^{hw} > \delta_{\text{fg}}]} \ell_{\text{bce}}(\hat{y}_i^{hw}, p_i^{hw}) \tag{29}$$

where $\mathbb{1}_{[A_i^{hw} > \delta_{\text{fg}}]}$ is a mask applied to ensure only regions associated with a normalized activation intensity higher than δ_{fg} are considered as *foreground* hints.

Figure 14 illustrates a few examples of saliency maps produced by the saliency detection model trained with C^{2}AM-H. An increase in quality of the maps is noticeable, when compared to the ones obtained by applying the *maximum* operation over CAMs.

Guiding Random Walk using Saliency Maps Originally, affinity maps — used in the training of the AffinityNet model [3] — are devised by applying the δ_{bg} and δ_{fg} thresholds over CAMs to determine core regions (likely depicting *bg* or *fg* regions, respectively). Going in a different direction, we propose a slight modification to this procedure that incorporates the saliency maps obtained from C^{2}AM-H: we leverage the saliency maps to more accurately determine *bg* regions in an image, and combine them with the confident *fg* regions to produce

Figure 13: Comparison between the different affinity maps obtained from RS269 trained with P-OC$_{+\text{LS}}$. From left to right: (a) images and ground-truth segmentation; (b) affinity labels devised from priors; (c) affinity labels refined with dCRF; and (d) affinity labels obtained using both C^{2}AM-H and dCRF.

the affinity maps.

Figure 13 illustrates affinity maps produced by both conventional and modified approaches (3rd and 4th columns, respectively): Many of the *background* regions, previously

Figure 14: Saliency maps generated by the PoolNet model, trained over saliency priors from C²AM-H (hints from P^f-NOC$_{+LS}$).

marked as *unknown* when considering CAMs, are now correctly assigned to *bg*. An increase in fidelity to semantic boundaries is also noticeable.

B.2.3 Quantitative Results

Table 7 illustrates mIoU measured at the end of each training epoch, considering various architectures and training strategies. For performance purposes, samples are resized to a common frame, and Test-Time Augmentation (TTA) is not employed. Hence, the intermediate measurements are estimations of the true scores (represented in the last column).

Table 7: The mIoU (%) values measured in each epoch over Pascal VOC 2012 *train* set, for each architecture (ResNeSt101 (RS101) and ResNeSt269 (RS269)) and training strategy (RandAugment (RA), Puzzle (P), and Puzzle-OC (P-OC)). Scores for $P^f$and P-OC were averaged among three distinct runs for increased stability.

Strategy	**E1**	**E2**	**E3**	**E4**	**E5**	**E6**	**E7**	**E8**	**E9**	**E10**	**E11**	**E12**	**E13**	**E14**	**E15**	**Max**	**TTA**
RS101 RA	48.7	48.2	**50.5**	50.4	49.1	49.5	49.2	49.8	41.0	48.9	48.7	49.4	48.9	49.2	49.2	50.5	54.8
RS269 RA	47.3	49.0	**49.3**	49.2	49.2	48.8	48.7	48.7	48.6	48.7	48.7	48.2	48.0	48.1	48.1	49.3	53.9
RS101 P	50.2	51.7	53.0	53.3	55.0	53.8	54.5	54.7	54.0	55.0	54.6	55.1	**55.6**	55.0	55.4	55.6	61.9
RS269 P	50.7	53.0	53.9	56.0	55.0	56.3	54.7	55.1	56.8	**57.6**	56.1	56.0	57.0	55.6	56.8	57.6	62.0
RS101 P^f	50.4	51.4	53.2	53.4	52.5	52.9	54.0	54.7	54.2	51.8	53.8	54.7	54.2	54.6	**54.9**	54.9	59.4
RS269 P^f	50.4	52.5	54.3	53.9	55.5	55.9	55.4	56.1	56.3	**57.0**	55.6	56.7	55.2	56.7	56.2	57.0	60.9
RS101 P-OC	49.6	50.3	51.5	51.8	52.5	51.5	49.0	49.9	53.2	52.5	53.4	54.2	54.9	55.5	**56.0**	56.0	59.1
RS269 P-OC	49.0	51.1	52.6	53.6	54.1	53.8	51.9	54.8	55.2	55.7	54.1	55.6	57.0	57.0	**57.4**	57.4	61.4
RS269 P-OC$_{+LS}$	50.6	52.5	53.5	54.3	53.9	55.0	55.2	55.3	56.4	56.1	55.8	56.2	55.9	57.5	**58.5**	58.5	61.8
RS269 P-NOC$_{+LS}$	50.6	52.2	53.5	55.6	56.3	56.5	57.2	55.8	57.1	57.6	58.7	**58.7**	58.6	58.6	58.5	**58.7**	**62.7**

RandAugment (RA) and Puzzle (P^f) present saturation on early epochs, and a significant

deterioration in mIoU for the following ones. On the other hand, Combining Puzzle and OC-CSE (P-OC) induces a notable increase in mIoU for all architectures, with performance peaking on the last epochs. On average, P-OC obtains 61.44% mIoU when TTA is used, lower than the original Puzzle (62.04%), but 0.55 p.p. above its fair counterpart (60.89%). Adding *label smoothing* to P-OC (P-OC$_{+\text{LS}}$) improves TTA score by 0.34 p.p. (61.77%). Finally, training OC (P-NOC$_{+\text{LS}}$) results in 62.67% mIoU (0.90 p.p. improvement).

Figure 15 displays the variance in overall mIoU of priors when the *fg* threshold is changed. P^{f} is only marginally better than the baseline, while P-OC and P-NOC display higher area under the curve, indicating that they are more robust to variations in the threshold.

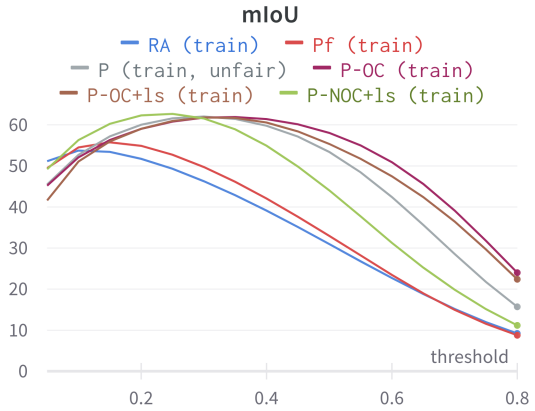


Figure 15: Curve of mIoU measured over Pascal VOC 2012 training set, considering multiple choices of threshold.

Table 8 contains the mIoU scores per class, for each one of the aforementioned models. Class and group statistics are also displayed for comparison purposes. Puzzle shows the best score for singleton and large classes, while adding OC-CSE induces a considerable score increase for small and mid-sized classes, as well as classes occurring in the room group.

Table 9 displays the mIoU results for saliency detection models trained from pseudo saliency maps produced by the *baseline* [84] (C^{2}AM) and C^{2}AM-H models, trained with additional *fg* hints. The combination of C^{2}AM with priors from RS269 P^{f} induces a score decrease when compared to the baseline RN50 P (0.23 p.p.) and RS269 P (1.18 p.p.). Conversely, RS101 RA and RS269 P-OC improve mIoU, with the latter achieving the highest mIoU observed (67.31%, 1.81 p.p. above the baseline).

Table 8: Scores (in IoU) measured over the Pascal VOC 2012 *train* set, per method. Class-specific properties, such as the avg. relative size (%S), class co-occurrence rate (%C) and avg. label set cardinality (L) are listed for inspection purposes.

Class	**Group**	**Size**	**%S**	**%C**	**L**	**RA**	**P**	**P^f**	**P-OC**	**P-OC$_{+LS}$**	**P-NOC$_{+LS}$**
bg	-	-	69.5	-	-	81.0	85.6	86.0	86.1	85.6	86.1
a.plane	singleton	mid	11.8	9.1	1.1	47.5	60.9	61.4	62.1	62.3	59.8
bicycle	traffic	small	6.4	76.9	2.2	32.2	41.2	38.6	44.6	45.0	39.9
bird	singleton	mid	11.8	11.4	1.2	49.5	69.7	71.4	63.6	62.8	68.6
boat	p-rel	small	10.8	32.1	1.4	40.8	45.2	51.3	50.9	43.9	48.5
bottle	bottle	small	9.5	70.1	2.3	49.0	56.9	56.0	59.2	65.9	65.9
bus	traffic	large	31.5	51.3	1.7	72.1	79.6	78.4	78.8	75.1	79.9
car	traffic	mid	15.5	61.7	1.9	62.6	74.2	70.4	72.5	74.6	75.9
cat	singleton	large	28.6	26.0	1.3	54.8	82.6	83.7	80.8	79.8	83.1
chair	room	small	10.6	87.8	2.4	30.7	28.9	27.2	21.6	23.1	27.0
cow	p-rel	mid	18.0	29.7	1.4	55.1	71.5	73.6	70.1	70.6	71.6
table	room	large	22.5	95.1	2.6	52.5	49.6	39.2	44.4	51.0	50.9
dog	p-rel	large	19.8	38.0	1.5	61.3	78.8	80.8	80.9	76.9	77.9
horse	p-rel	large	19.1	47.1	1.6	55.9	67.7	69.0	69.6	69.8	70.1
m.bike	traffic	large	19.6	56.8	1.7	67.8	74.4	73.2	78.2	76.6	73.7
person	person	mid	15.2	83.0	2.1	63.6	57.0	50.3	67.1	70.2	54.4
p.plant	room	small	11.2	63.4	2.1	46.8	57.8	56.3	45.1	57.2	57.2
sheep	p-rel	large	19.7	19.0	1.3	55.3	75.0	75.9	78.2	73.9	72.2
sofa	room	large	21.6	80.6	2.4	50.0	40.9	35.5	40.1	34.8	44.4
train	p-rel	large	26.6	20.5	1.2	63.9	68.9	68.2	63.4	50.9	68.1
tv	room	mid	15.5	65.1	2.1	38.6	36.5	33.8	43.6	49.8	42.0
overall			19.8	51.2	1.8	53.9	62.0	61.0	61.9	61.9	62.7
small			9.7	66.1	2.1	39.9	46.0	45.9	44.3	47.0	47.7
mid			14.6	43.3	1.6	52.8	61.6	60.1	63.2	65.1	62.0
large			23.2	48.3	1.7	59.3	68.6	67.1	68.3	65.4	68.9
singleton			17.4	15.5	1.2	50.6	71.1	72.2	68.8	68.3	70.5
p-rel			19.0	31.1	1.4	55.4	67.8	69.8	68.8	64.3	68.1
room			16.3	78.4	2.3	43.7	42.7	38.4	38.9	43.2	44.3
traffic			18.3	61.7	1.9	58.7	67.3	65.1	68.5	67.8	67.4
ρ %S			100.0	-18.0	-23.4	70.3	55.6	48.4	50.7	38.3	55.5
ρ %C			-18.0	100.0	97.6	-22.9	-60.6	-71.0	-57.3	-42.9	-58.0
ρ L			-23.4	97.6	100.0	-32.9	-66.0	-75.8	-65.4	-49.7	-61.9

To isolate the contribution of the saliency maps over the result, we also evaluate models using the ground-truth (supervised) segmentation annotations (GT) as priors. I.e., the prediction for a given pixel in the image is considered correct if that pixel was predicted as

salient and it is annotated with class c. Conversely, a pixel annotated with c and predicted as non-salient (or annotated as *bg* and predicted as salient) counts as a *miss*.

In this evaluation setup, the baseline (C²AM RN50) scored 65.03% mIoU, while the best strategy (C²AM-H, using hints from RS269 P-OC$_{+LS}$) achieves 71.70% mIoU (a 6.67 p.p. increase). Replacing the architecture of C²AM (C²AM RS269), or using hints to train the RN50 architecture (C²AM-H RN50) produced mixed results: a 2.02 p.p. score reduction for the former, and a 1.39 p.p. score increase for the latter. We hypothesize this has occurred due a representation deficit created when training with a reduced batch size, and to an inability of the RN50 architecture to produce more detailed maps. Finally, the best strategy (P-OC) achieves 69.22% mIoU when combined with real priors, with only a marginal difference to P-NOC.

Table 9: The ablation study for C²AM-H over VOC12 training set. Scores are reported in mIoU (%), considering both priors (P%) and maps refined with PoolNet (R%).

Method	**B.bone**	**Hints**	**CAM**	**P%**	**R%**
C²AM	RN50	-	RN50 P	56.6	65.5
C²AM	RN50	-	RS101 RA	61.2	66.2
C²AM	RN50	-	RS101 P-OC	60.4	66.6
C²AM	RN50	-	RS269 P	60.3	66.5
C²AM	RN50	-	RS269 P^{f}	59.1	65.3
C²AM	RN50	-	RS269 P-OC	60.8	67.3
C²AM	RN50	-	RS269 P-OC$_{+LS}$	61.2	67.2
C²AM	RN50	-	GT	63.4	65.0
C²AM	RS269	-	GT	61.4	-
C²AM-H	RN50	RS269 P-OC	GT	64.8	-
C²AM-H	RS101	RS269 P-OC	GT	69.6	-
C²AM-H	RS269	RS269 P-OC	GT	69.9	70.9
C²AM-H	RS269	RS269 P-OC$_{+LS}$	GT	70.3	71.7
C²AM-H	RS269	RS269 P-NOC$_{+LS}$	GT	70.2	71.3
C²AM-H	RS269	RS269 P-OC	RS101 RA	66.5	66.2
C²AM-H	RS269	RS269 P-OC	RS101 P-OC	66.7	67.9
C²AM-H	RS269	RS269 P-OC	RS269 P-OC	66.8	68.6
C²AM-H	RS269	RS269 P-OC	RS269 P-OC$_{+LS}$	67.3	68.8
C²AM-H	RS269	RS269 P-OC$_{+LS}$	RS269 P-OC$_{+LS}$	**67.3**	**69.2**
C²AM-H	RS269	RS269 P-OC$_{+LS}$	RS269 P-NOC$_{+LS}$	67.2	69.1
C²AM-H	RS269	RS269 P-NOC$_{+LS}$	RS269 P-NOC$_{+LS}$	67.2	68.4

Table 10 describes the scores obtained throughout the different stages of training. The utilization of saliency maps (P-OC C²AM-H) increases the mIoU of the pseudo segmentation

Table 10: Ablation studies of pseudo segmentation masks, measured in mIoU (%) over VOC12 training and validation sets.

Method	**+LS**	**+C²AM-H**	**+NOC**	***train* (%)**	***val* (%)**
P				73.74	72.31
P^f				71.35	70.67
P-OC				73.50	72.08
P-OC	✓			71.45	70.15
P-OC		✓		**73.90**	72.53
P-OC	✓	✓		73.07	72.14
P-OC	✓		✓	73.31	72.83
P-OC	✓	✓	✓	73.59	**73.37**

masks by 0.40 p.p., and a DeepLabV3+ [12] model trained over those result in 74.34% and 71.38% mIoU over the VOC12 training and validation sets, respectively. Finally, training the Ordinary Classifier (+NOC) results in a 0.31 p.p. decrease in mIoU over the training subset, while increasing validation mIoU by 0.84 p.p..

Examples of segmentation maps predicted by the DeepLabV3+ model are illustrated in Figure 16. High coverage and sensitivity to the semantic boundaries of objects is noticeable for the classes *person*, *dog*, *cat*, *horse*, and *car*. Conversely, we observe failure segmentation cases for the classes *table* and *chair* (with low coverage), and classes *train* and *sofa*, in which their respective segmentation maps extrapolate their boundaries onto the background.

Table 11 shows IoU scores (per-class) obtained by the DeepLabV3+ model, when trained with pseudo segmentation masks created from P-OC and P-NOC. Classes associated with (i) small objects, (ii) complex and highly-detailed semantic boundaries, and (iii) often appearing in cluttered scenarios often present lower than average mIoU scores (e.g., bycicle, chair). Conversely, classes associated with (i) large objects and singletones and (ii) simple convex semantic boundaries often present high IoU scores (e.g., airplane, bird, bus, cat, dog, sheep).

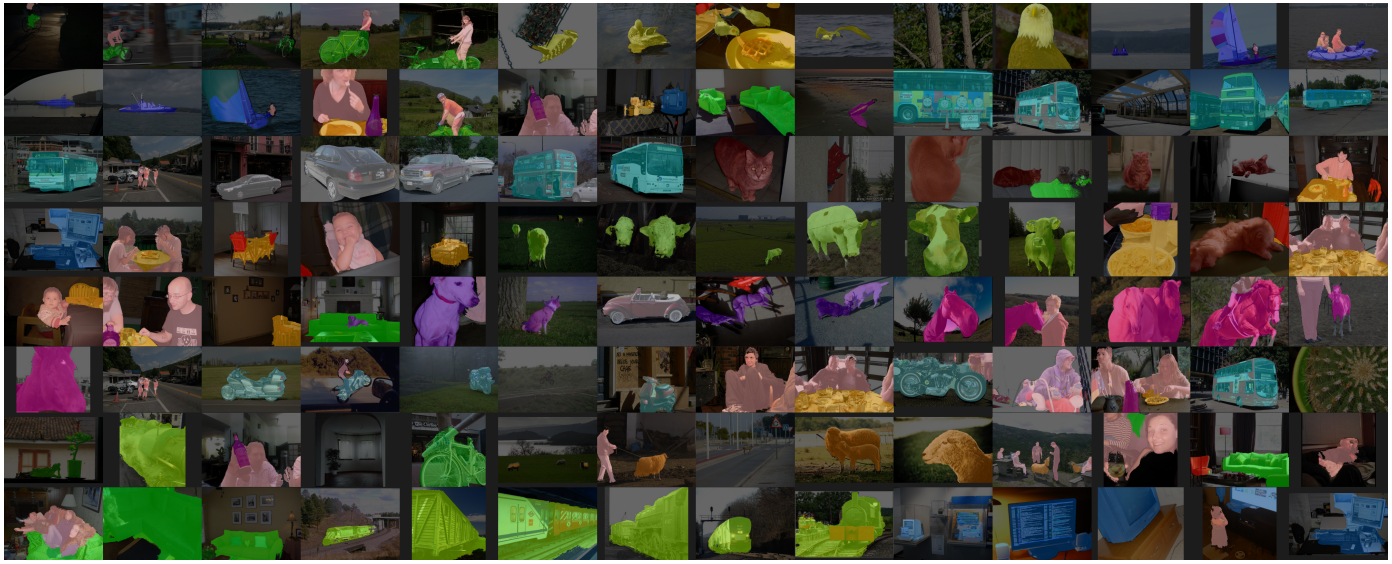


Figure 16: Segmentation results by DeepLabV3+, trained with priors obtained from P-OC and refined with C^{2}AM-H and RW.

Table 11: Intersection over Union (IoU %) for each class in the Pascal VOC 2012 testing dataset.

	bg	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbk.	person	plant	sheep	sofa	train	tv	avg.
AffinityNet	88.2	68.2	30.6	81.1	49.6	61.0	77.8	66.1	75.1	29.0	66.0	40.2	80.4	62.0	70.4	73.7	42.5	70.7	42.6	68.1	51.6	61.7
OC-CSE	90.2	82.9	35.1	86.8	59.4	70.6	82.5	78.1	87.4	30.1	79.4	45.9	83.1	83.4	75.7	73.4	48.1	**89.3**	42.7	60.4	52.3	68.4
AMN	90.7	82.8	32.4	84.8	59.4	70.0	86.7	83.0	86.9	30.1	79.2	56.6	83.0	81.9	78.3	72.7	52.9	81.4	59.8	53.1	**56.4**	69.6
ViT-PCM	91.1	**88.9**	**39.0**	87.0	58.8	69.4	89.4	85.4	89.9	30.7	82.6	**62.2**	85.7	83.6	79.7	**81.6**	52.1	82.0	26.5	**80.3**	42.4	70.9
MCT-Former	**92.3**	84.4	37.2	82.8	60.0	72.8	78.0	79.0	89.4	**31.7**	84.5	59.1	85.3	**83.8**	79.2	81.0	53.9	85.3	**60.5**	65.7	57.7	71.6
P-OC (ours)	91.6	86.7	38.3	**89.3**	61.1	**74.8**	92.0	**86.6**	89.9	20.5	**85.8**	57.0	90.2	83.5	**83.4**	80.8	**68.0**	87.0	47.1	62.8	43.1	72.4
P-NOC (ours)	91.4	86.7	35.2	87.8	**62.9**	71.6	**93.0**	86.3	**92.3**	30.4	**85.8**	60.7	**91.7**	81.7	82.7	66.3	65.9	88.8	48.7	72.5	44.5	**72.7**