Bag-level classification network for infrared target detection

Connor McCurley, Daniel Rodriguez, Chandler Trousdale, Arielle Stevens, Anthony Baldino, et al.

Connor H. McCurley, Daniel Rodriguez, Chandler Trousdale, Arielle Stevens, Anthony Baldino, Eugene Li, Isabella Perlmutter, Alina Zare, "Bag-level classification network for infrared target detection," Proc. SPIE 12096, Automatic Target Recognition XXXII, 1209603 (31 May 2022); doi: 10.1117/12.2618325

SPIE.

Event: SPIE Defense + Commercial Sensing, 2022, Orlando, Florida, United States

Bag-level Classification Network for Infrared Target Detection

Connor H. McCurley, Daniel Rodriguez, Chandler Trousdale, Arielle Stevens, Anthony Baldino, Eugene Li, Isabella Perlmutter, and Alina Zare

Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611, USA

ABSTRACT

Aided target detection in infrared data has proven an important area of investigation for both military and civilian applications. While target detection at the object or pixel-level has been explored extensively, existing approaches require precisely-annotated data which is often expensive or difficult to obtain. Leveraging advancements in weakly supervised semantic segmentation, this paper explores the feasibility of learning a pixel-level classification scheme given only image-level label information. Specifically, we investigate the use of class activation maps to inform feature selection for binary, pixel-level classification tasks. Results are given on four infrared aided target recognition datasets of varying difficulty. Results are quantitatively evaluated using common approaches in the literature.

Keywords: target detection, weakly supervised semantic segmentation, weak learning, multiple instance learning, imprecise labels, infrared, feature selection, class activation map

1. INTRODUCTION

Target detection is a paramount area of research in the field of remote sensing which aims to locate an object or region of interest while suppressing unrelated objects and information.[1,2] Binary target detection can be formulated as a two-class classification problem where samples belonging to a class of interest are discriminated from a background, or non-target, distribution.[1–3] Common applications require targets to be classified at the pixel-level, and traditional supervised learning approaches require extensive amounts of highly precise, pixel-level groundtruth to guide algorithmic training. However, acquiring large quantities of accurately labeled training data can be expensive both in terms of time and resources, and in some cases, may even be infeasible to obtain.[4,5] Because of these limitations, achieving pixel-level classification, or *semantic segmentation*, in realistic environments can be challenging.[6]

This problem has motivated the exploration of learning from alternative types of labels, deemed *weak*, *imprecise* or *uncertain*.[7–14] As such, approaches for *weakly supervised semantic segmentation* (WSSS) have been explored using bounding boxes,[15,16] scribbles,[17–22] points,[23–26] and image-level[16] labels that are less informative than pixel-level labels, but are readily available in large quantities or easily obtained due to their low annotation costs. Figure 1 demonstrates different types of imprecise

Further author information: (Send correspondence to C.H.M.)
C.H.M.: E-mail: cmccurley@ufl.edu
A.Z.: E-mail: azare@ece.ufl.edu

Automatic Target Recognition XXXII, edited by Riad I. Hammoud, Timothy L. Overman, Abhijit Mahalanobis, Kristen Jaskie, Proc. of SPIE Vol. 12096, 1209603
 · 0277-786X · doi: 10.1117/12.2618325

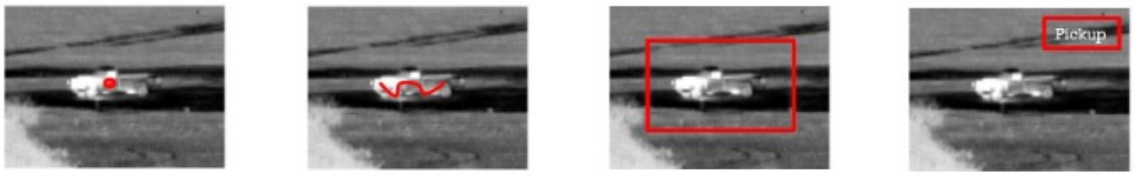

Figure 1: Examples of weakly-labeled infrared imagery. The images demonstrate various forms of weak groundtruth for a pickup truck captured with a mid-wave infrared camera. The images show spot, scribble, imprecise bounding-box, and image-level labels, respectively.

labels. This work considers the case of learning pixel-level segmentations from image-level labels. Approaches in this line of research often incorporate additional evidence to infer location and shape information which is absent in the label information.[6] A popular localization mechanism is the *Class Activation Map* (CAM),[27] which estimates regions in an input image which contribute to the estimated class label by analyzing the activations of hidden units in the output layers of a deep convolutional neural network (DCNN).

This paper investigates pixel-level target classification in infrared data from image-level labels. The contributions of this paper are threefold:

- Classifiers were trained to predict whether or not infrared images contained target pixels (i.e. bag-level classification). Experiments were conducted on four datasets of varying difficulty, as well as varying levels of label imprecision.
- WSSS was achieved by thresholding CAMs estimated in various hidden layers of the trained image-level classification network.
- Experiments using CAMs as pseudo-groundtruth for activation map feature selection were conducted and tested in pixel-level classification.

To provide insight into future directions for WSSS in infrared imagery, this work utilized post-hoc attention under the paradigm of multiple instance learning to explore WSSS. Specifically, this work shows that a DCNN is capable of distinguishing between positive and negative bags, leading to the presumption that features which can discriminate instance (pixel) labels can be abstracted from the information contained in the model. Class activation maps were explored as a baseline for performing WSSS from the learned features of the trained bag-level classifiers. Finally, feature reduction/extraction from the trained models was explored as a way to abstract instance-level classification information from the bag-level classification models. Each method was applied on hold-out test data to evaluate the effectiveness of the segmentation techniques. Quantitative results are given as overall classification accuracy for bag-level classification and mean intersection-over-union (mIoU) for semantic segmentation.

An overview of the investigated approach is shown in Figure 2. The training method is performed in three stages, each of which adheres to MIL constraints. Stage (a) trains a bag-level classification network. Stage (b) estimates a class activation map for the inferred bag-level label. Stage (c) up-samples and concatenates the activation feature maps from the bag-level classification network,

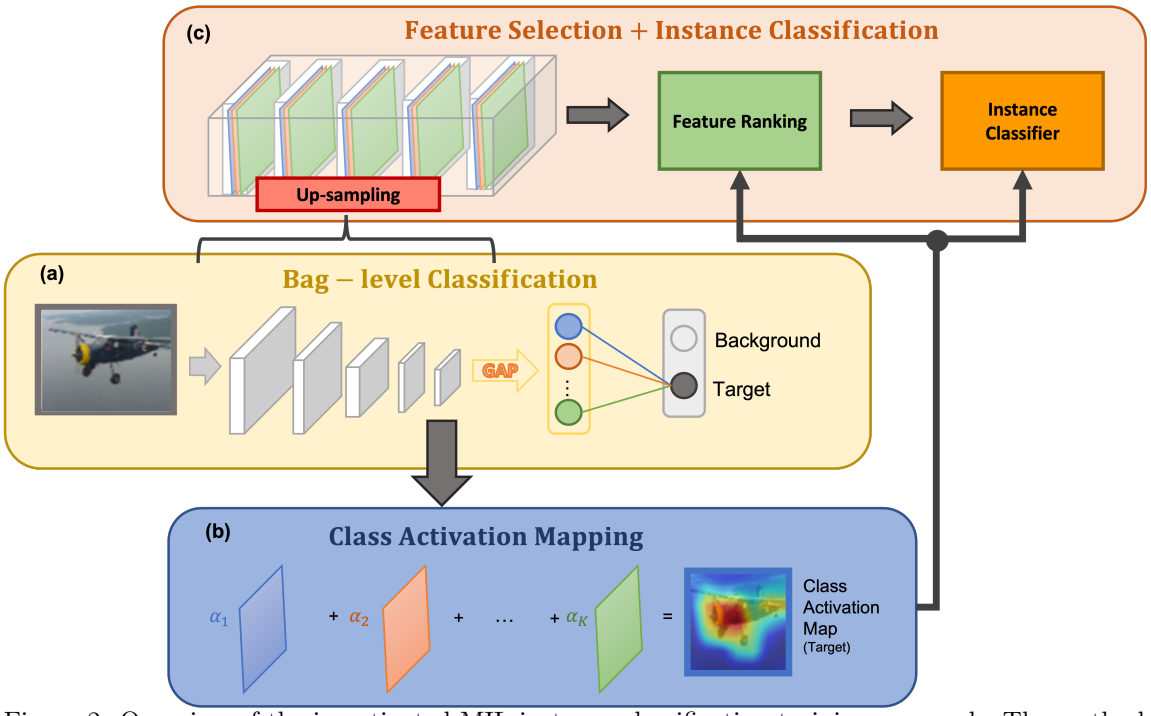

Figure 2: Overview of the investigated MIL instance-classification training approach. The method is performed in three stages. Stage (a) trains a bag-level classification network. Stage (b) estimates a class activation map for the inferred bag-level label for a training image. Stage (c) up-samples and concatenates the activation feature maps from the bag-level classification network. This provides a feature vector at every pixel/instance. The corresponding class activation map from Stage (b) is used as a pseudo-label to allow for feature ranking. Post-ranking, an instance-level classifier is trained, using the CAM as groundtruth. All three training stages maintain MIL constraints.

which provides a feature vector at every pixel/instance. The corresponding class activation map from Stage (b) is used as a pseudo-label and feature ranking is performed. An instance-level classifier is trained using the ranked features and corresponding CAM as groundtruth. In this manner, an instance-level classifier is estimated from bag-level labels. Each stage of the approach is described in detail in Section 3.

The rest of this paper is organized as follows. Section 2 describes the mid-wave infrared data imagery used in this paper. Section 3 describes the bag-level classifier training as well as the process used for CAM estimation and MIL feature selection. Next, Section 4 details the aforementioned experiments and summarizes results. Final thoughts and conclusions are given in Section 5.

2. DESCRIPTION OF DATA

The data used in this work consists of mid-wave infrared (MWIR) video captures of both moving and non-moving civilian and military vehicles at various ranges and aspects. Each video can be considered as a collection of frames taken at the corresponding sensor's sampling rate. Datasets were each processed and broken into subsets amenable for learning with MIL. As shown in Figure

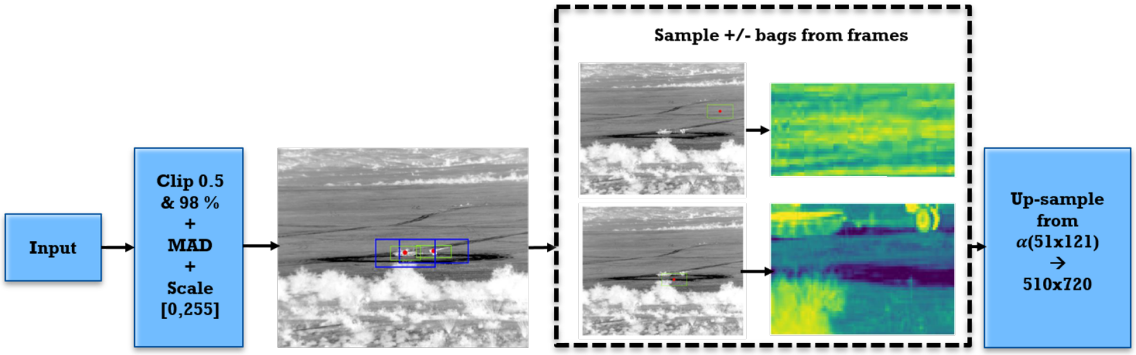

Figure 3: Frame pre-processing and bag sampling pipeline.

3, each gray-scale image frame, $I \in \mathbb{R}^{+(510\times720)}$, was clipped at 0.5% and 98%. The images then underwent normalization by median absolute deviation (MAD), defined as

$$\text{MAD} = \text{median}(|I - \text{median}(I)|), \tag{1}$$

and were scaled to $[0, 255]$. Images in the datasets were originally annotated with bounding boxes. In order to analyze the data under MIL, sub-images were extracted to create sets of bags (see Section 3.1), where positive bags each contained at least one pixel on target and negative bags contained only background pixels. Following these constraints, sub-images were sampled such that negative bags had no overlap with target bounding boxes and positive bags had at least 25% overlap with a target bounding box. Sub-samples were taken in scalar values of (51×121), which corresponds to the largest target present in the datasets. All sub-sampled images were up-sampled with bilinear interpolation to the original frame size of (510×720).

As can be seen in Figure 4, bags were constructed to represent various levels of groundtruth imprecision. This was done by changing the ratio of background to target pixels in the sub-sampled image chip. Essentially, canonical bags were constructed from the provided bounding box annotations where the majority of pixels fall target. Ratios α = 1-3 increase the sample size as a scalar value of (51×121). As the scalar value increases, the ratio of background to target pixels in the image also increases. Training and testing on bags consisting of different ratios of target and non-target instances provides a way to capture the ability of a model to abstract pixel-level label information from the bags and corresponding bag-level labels.

3. METHODOLOGY

In this work, the multiple instance learning WSSS problem is explored. First, a deep convolutional neural network is trained to predict whether an image is a positive or negative bag. After successful training, class activation maps are computed from the features of the network. The CAMs provide weak target localization information which is used to inform pixel-level classification. This chapter provides technical details of the methods explored in this work. An overview of multiple instance learning is provided along with details of the bag classification network. A summary of class activation maps is given and details of a novel feature selection approach using CAM pseudo-groundtruth are discussed.

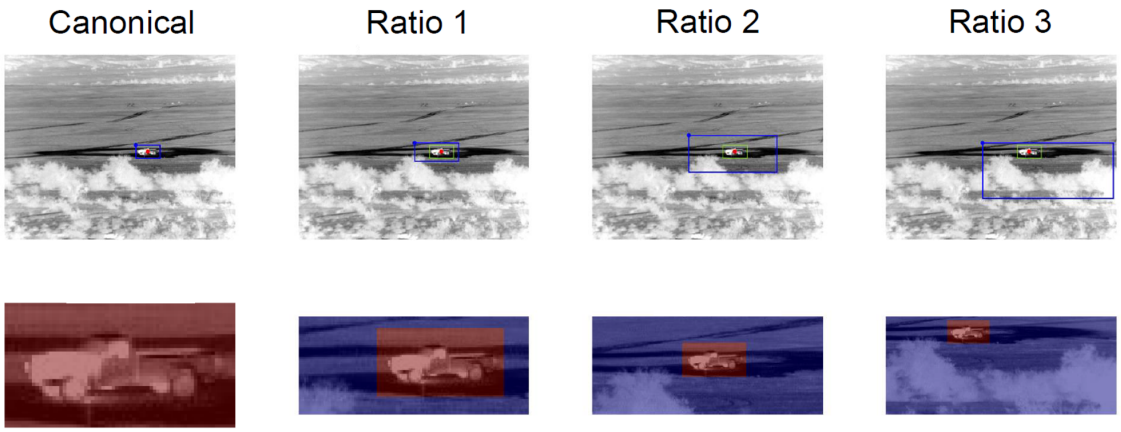

Figure 4: Demonstration of positive bag sampling for various levels of imprecision. In the top row, green boxes represent the bounding box annotation, while blue boxes represent the frame sub-sample capture. Annotation bounding boxes are represented in red in the bottom row of images. Canonical bags are constructed from the provided bounding box annotations and the majority of pixels fall on target. Ratios 1-3 increase the sample size as a scalar value of (51×121). As the scalar value increases, the ratio of background to target pixels in the image also increases.

3.1 Multiple Instance Learning

Multiple Instance Learning (MIL) was originally proposed by Dietterich[28] as a method to handle inherent observation difficulties associated with drug activity prediction. This problem, among others, fits well into the framework of MIL where training labels are associated with sets of data points, called *bags* instead of individual samples, or *instances*. Under the *standard MIL assumption*, a bag is given a "positive" label if it is known that *at least one* sample in the set represents pure or partial target. Alternatively, a bag is labeled as "negative" if does not contain any positive instances.[13] Let $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N] \in \mathbb{R}^{D \times N}$ be training data where D is the dimensionality of an instance, $\boldsymbol{x}_n$, and N is the total number of training instances. The data is grouped into K *bags*, $\boldsymbol{B} = \{\boldsymbol{B}_1, \ldots, \boldsymbol{B}_K\}$, with associated binary bag-level labels, $\mathcal{L} = \{L_1, \ldots, L_K\}$ where

$$L_k = \begin{cases} +1, & \exists \boldsymbol{x}_{kn} \in \boldsymbol{B}_k^+ \ni l_{kn} = +1 \\ 0, & l_{kn} = -1 \quad \forall \boldsymbol{x}_{kn} \in \boldsymbol{B}_k^- \end{cases}, \tag{2}$$

$\boldsymbol{x}_{kn}$ denotes the n^{th} instance in positive bag $\boldsymbol{B}_k^+$ or negative bag $\boldsymbol{B}_k^-$, and $l_{kn} \in \{0, +1\}$ denotes the instance-level label on instance $\boldsymbol{x}_{kn}$. Figure 5 demonstrates the concept of MIL bags. MIL has recently been explored for target detection using a variety of remote sensing modalities.[3,29–34] The objective of learning under MIL in this work is, given only bag-level label information, to fit a model which can perform bag (image) and instance (pixel)-level classification.

3.2 Bag-Level Classification

In this work, a bag represents an image $\boldsymbol{B} \triangleq I$, where $I \in \mathbb{R}^{u \times v \times w}$. Thus, a bag is a collection of feature vectors (instances) at every pixel spatial location, (u, v). Each training image I_k is paired with a label $L_k \in \{0, 1\}$, where $L_n = 0$ is assigned if every pixel in the image belongs to the background class (i.e. $l_{kn} = 0 \, \forall \boldsymbol{x}_{kn} \in \boldsymbol{B}_k^-$), and $L_n = 1$ if at least one pixel in the image belongs to

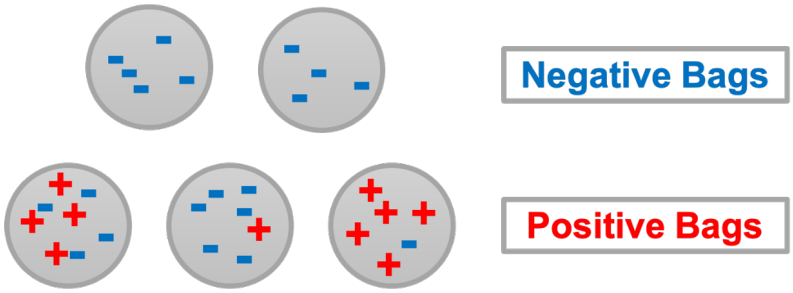

Figure 5: Illustration of example bags under the multiple instance learning framework. Red "plus signs" denote positive instances and blue "negative signs" represent negative instances. The two bags on the top row are labeled "negative" because they only contain negative instances. The three bags on the bottom row are "positive" because they each contain at least one positive instance.

the target class ($\exists \boldsymbol{x}_{kn} \in \boldsymbol{B}_k^+ \ni l_{kn} = 1$). While each bag is given a label according to the standard MIL assumption, the labels of individual instances in positive training bags are unknown.

A binary classification network was trained on the images with image-level labels (Stage (a), Figure 2). Consider one-hot encoded labels $\boldsymbol{L}_k = [L_{k0}, L_{k1}]$ where $\boldsymbol{L}_k = [0, 1]$ for a positive bag and $\boldsymbol{L}_k = [1, 0]$ for a negative bag. The goal of the binary classification network is to estimate the probability $p(\boldsymbol{B}_k, \boldsymbol{\theta}) = [p_0(\boldsymbol{B}_k, \boldsymbol{\theta}), p_1(\boldsymbol{B}_k, \boldsymbol{\theta})]$ that a bag belongs to the target or background class. A softmax output is applied to the output of the network such that $p_0(\boldsymbol{B}_k, \boldsymbol{\theta}), p_1(\boldsymbol{B}_k, \boldsymbol{\theta}) \geq 0$ and $p_0(\boldsymbol{B}_k, \boldsymbol{\theta}) + p_1(\boldsymbol{B}_k, \boldsymbol{\theta}) = 1$. Cross-entropy loss, $\mathcal{L}_{cls}(\boldsymbol{B}; \boldsymbol{\theta}, \boldsymbol{L})$, was used on the image classification network to minimize the error between each predicted image-level label and the true class:

$$\min_{\boldsymbol{\theta}} \mathcal{L}_{cls}(\boldsymbol{B}; \boldsymbol{\theta}, \boldsymbol{L}) = \frac{1}{K} \sum_{k=1}^{K} \sum_{i=1}^{2} L_{ki} \log p_{bag}^{i}(\boldsymbol{B}_k; \boldsymbol{\theta}) \tag{3}$$

where $p_{bag}^{i}(\boldsymbol{B}_k; \boldsymbol{\theta})$ is the i^{th} element of the softmax output layer for the bag classification network parameterized by $\boldsymbol{\theta}$.

3.3 Class Activation Mapping

A *Class Activation Map* (CAM)[35] is a post-hoc method for visualizing attention in convolutional neural networks (CNN) which has been used extensively in the WSSS literature.[36–49] Essentially, a CAM for a particular class label indicates the discriminative image regions used by the CNN to identify the category. As shown if Figure 6, the neural network structure for a CAM commonly consists of stacked convolutional layers, a global pooling layer, a fully connected layer (fc), and the output layer. Formally, let f denote the image classifier parameterized by $\boldsymbol{\theta}$. For a given image $I \in \mathbb{R}^{u \times v \times w}$, the predicted score y^c of the target category c before input to the softmax is given by

$$y^c = f^c(I, \boldsymbol{\theta}). \tag{4}$$

Let $\boldsymbol{A}^k \in \mathbb{R}^{i \times j}$ be the k-th feature map in the final convolutional layer. The input to the softmax is the sum of the activations scaled by their relative importances, α_k^c, toward each class label. The

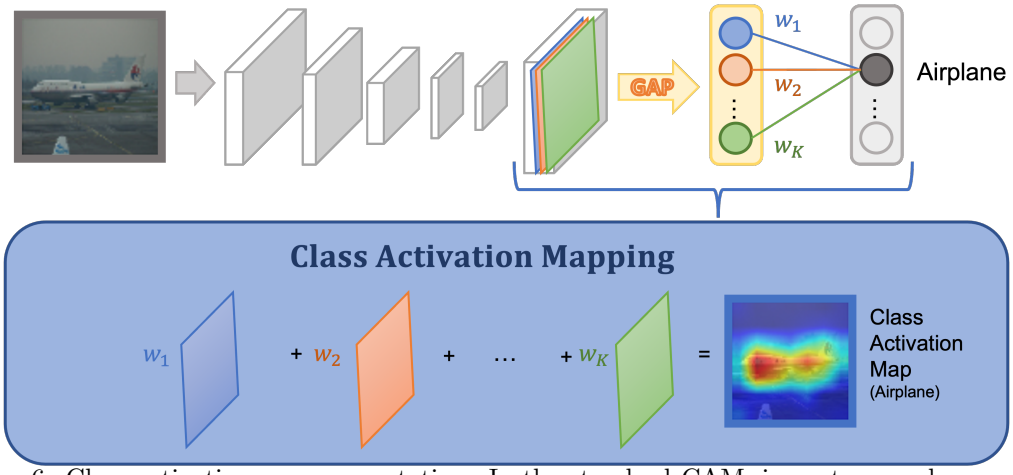

Figure 6: Class activation map computation. In the standard CAM, importance values are given as the weights between global average pooled feature maps and their weights toward a particular class in the fully-connected layer.

localization map for CAM, $\boldsymbol{L}^c_{\text{CAM}}$, for class c is obtained by applying a ReLU operation on the summation to remove negative responses

$$\boldsymbol{L}^c_{\text{CAM}} = \text{ReLU}\left(\sum_k \alpha^c_k \boldsymbol{A}^k\right). \tag{5}$$

For the standard CAM, the importance weights, $\alpha^c_k = w^c_k$, are given as the weights in the fully connected layer contributing to a particular class score. Activation maps are up-sampled using bilinear interpolation to the match the input and each element in the heatmap is scaled to $[0, 1]$.

Grad-CAM[50] and *Grad-CAM++*[51] generalize CAM[35] by allowing for computation in any CNN architecture. In these approaches, activation maps are weighted by their average gradients. The gradient of prediction score y^c with respect to local spatial location (i, j) in feature map $\boldsymbol{A}^k$ is given as

$$g^{kc}_{ij} = \frac{\partial y^c}{\partial \boldsymbol{A}^k_{ij}}. \tag{6}$$

Grad-CAM obtains the channel-wise importance weighting by averaging the gradients over all locations in the feature map as

$$\alpha^c_k = \frac{1}{N}\sum_i \sum_j g^{kc}_{ij}, \tag{7}$$

where N is the number of spatial locations in feature map $\boldsymbol{A}_k$. *Grad-CAM++* is essentially the same as Grad-CAM, except that it uses second order gradient information to determine the feature importance weights. As a result, Grad-CAM++ has been shown to produce better object localization ability when multiple instances occur in the same image. Both Grad-CAM and Grad-CAM++ assign a single weight for each spatial location in an activation feature map. However, in shallow

layers, the variances of activation maps are often large, meaning a global weight cannot adequately represent the importance of different spatial locations toward particular categories.

LayerCAM[52] addresses this problem by assigning an importance weight to each spatial location in a feature map. Formally, the importance weight for spatial location (i, j) in the k-th feature map can be written as

$$\alpha_{ij}^{kc} = \text{ReLU}(g_{ij}^{kc}). \tag{8}$$

To obtain the class activation map for a particular layer, LayerCAM multiplies the activation value of each location by its importance weight

$$\hat{A}_{ij}^{kc} = \alpha_{ij}^{kc} \cdot A_{ij}^{kc}. \tag{9}$$

As with alternative approaches, the results $\hat{A}^k$, are combined linearly along the channel dimension to obtain the final CAM

$$\boldsymbol{L}_{\text{LayerCAM}}^{c} = \text{ReLU}\left(\sum_k \hat{\boldsymbol{A}}^k\right). \tag{10}$$

LayerCAM has been shown to generate reliable class activation maps in shallow layers, which capture fine-grained localization information.

Score-CAM[53] rids itself of dependence on gradients by finding importance weights of activation maps by a forward passing score in the network. Essentially, each activation map is used as a mask on the input image and the importance weights for the c classes are given as the output scores for the masked image.

Ablation-CAM[54] attempts to remedy the problem of diminishing gradients by avoiding them entirely. In their work, Desai et al.[54] showed that removing certain feature map units had a severe impact on the accuracy of certain classes. As a result, Ablation-CAM considers the performance drop to be an indicator of feature importance. The importance weight for activation feature map, $\boldsymbol{A}^k$, is computed as the performance drop between the output score y^c, and the output score with the k-th activation map removed, denoted as y_k^c. The importance weights are formally given as

$$\alpha_k^c = \frac{y^c - y_k^c}{y^c}. \tag{11}$$

Similarly to Score-CAM, Ablation-CAM uses forward class activation information to visualize attention in convolutional neural networks. These approaches have the advantage of using the inherent flow through a neural network, instead of relying on gradients which lose spatial information through pooling and magnitude information through activation functions (i.e. ReLU).

Eigen-CAM[55] is a class non-discriminative approach for identifying salience information in the input space. Similarly to Score-CAM and Ablation-CAM, Eigen-CAM does not rely on gradients, but obtains a saliency map by projecting the input image onto the first eigenvector of the convolutional feature map weights at a particular layer. This approach has the benefit of providing salience information irrespective of model accuracy. Additionally, it has been shown that Eigen-CAM is more robust to adversarial noise than alternative CAM approaches in the literature.

In this work, CAMs were computed from the trained bag-level classifier (Stage (b), (Figure 2)). The CAMs were binarized across a range of confidence thresholds to investigate the ability of the

network to inherently infer pixel-level semantics from image-level labels. Additionally, CAMs were explored as pseudo-labels for feature selection. While all methods were trained using imprecise labels, they were tested on a small subset of test data with accompanying pixel-level groundtruth.

3.4 Feature Selection from CAM Pseudo-groundtruth

Feature selection is a popular and straightforward approach for dimensionality reduction. Feature selection techniques define a smaller feature set by selecting a subset of the original features. There are three primary categories of feature selection techniques: *filtering* acts as a preprocessing step to construct an independent feature set before a classifier is constructed, *wrapper* techniques use the performance of the classifier as a fitness function on the set of features, and *embedded* methods include feature selection as part of the classifier's optimization objective (i.e. sparsity constraints).[56] *Feature ranking* is an appealing filtering approach for its speed and simplicity. Essentially, feature ranking approaches evaluate a fitness function on each feature, independently. The features are then sorted by their scores and the top K' are selected. Algorithm 1 depicts the feature ranking scheme used in this work to select activation maps from the trained bag-level classifier for instance-level classification. All activation maps are extracted from the trained classification network and up-sampled to the input dimensionality (Stage (c), Figure 2). The concatenation of all up-sampled activations provides a feature set which can be exploited for pixel-level classification. Let $\boldsymbol{A}$ be the set of up-sampled activations for an input image and let $\boldsymbol{L}_{\text{CAM}}$ be the set of corresponding class activation maps computed on the predicted image-level labels. The implemented feature ranking approach uses mIoU as the fitness function between a single activation map and the inferred class activation map. The approach maintains weak learning by setting the pseudo pixel-level labels as the class activation maps computed for the inferred bag label. There are implicit assumptions that the predicted bag labels are correct, and that the computed CAMs adequately define the targets at the pixel-level. Once features have been selected, a logistic regression classifier is trained to predict thresholded CAM pseudo-labels. The classification model with reduced feature set is then tested on the hold-out test data to evaluate the ability of the model trained on weak labels to compute accurate pixel-level segmentations.

A trade-off of using feature ranking is that, if multiple features are needed to discover a correlation, they will not be considered since each feature is evaluated independently. An alternative would be to use a method that selects features sequentially, such as forward or backward feature selection where features are added to the set one at a time depending on their fitness with the current set of features. Compared to feature ranking, which has a linear run time (K'), a forward feature selection tends to be polynomial in the number of features since each new feature added requires re-evaluation of the scoring function for every feature not already included in the set.

Algorithm 1 Feature Ranking

Input: Dataset $\boldsymbol{X} = \{\boldsymbol{X}_1, \ldots, \boldsymbol{X}_N\}, \boldsymbol{L}_{\text{CAM}} = \{\boldsymbol{L}_{CAM_1}, \ldots, \boldsymbol{L}_{CAM_K}\}$, scoring/fitness function $\mathcal{S}$, feature set $\boldsymbol{A} = \{\boldsymbol{A}_1, \ldots, \boldsymbol{A}_K\}$, new dimensionality K'

$V \leftarrow [\,]$
for feature $\boldsymbol{A}_k$ in feature set $\boldsymbol{A}$ **do**
    $V[k] \leftarrow (\mathcal{S}(\boldsymbol{X}, \boldsymbol{L}_{\text{CAM}}, \boldsymbol{A}_k), \boldsymbol{A}_k)$
end for
$V \leftarrow \textsc{SortDecreasing}(V)$
return $[\boldsymbol{A}_k$ for $\boldsymbol{A}_k$ in $V[1, \ldots, K']]$

4. EXPERIMENTS AND RESULTS

4.1 Data Description

Experiments were conducted on MWIR data taken from four distinct collection sites. The publicly-available DSIAC MS-003-DB Algorithm Development Database[57] is considered the easiest in the group, because targets in this set have very little occlusion and only collections from nighttime were used. It was anticipated that data from this site would provide the best bag-level performance and would be a good indicator of the possibility of learning pixel-level features from bag-level classification models. Sites "Y", "H", and "B" followed in assumed levels of difficulty. This was inferred from the targets included in the datasets, the levels of occlusion, and the various aspects to which the targets were visible. Also these three datasets included MWIR data collected at both day and night. The total numbers of image frames in each dataset are shown in Table 2. Only image-level labels were given to training and validation sets. Test sets were given hand-segmented pixel-level annotations for evaluating WSSS. Each split of the data contained equal numbers of positive/negative bags. Bag-level prediction was investigated for all four datasets, including all levels of bag imprecision (i.e. canonical/ratio 0 - ratio 3). To determine the role of positive bag construction (i.e. number of target versus background pixels) in inference, a bag-level classifier was also trained for each level of data imprecision for site "H". Pixel-level segmentation was evaluated on the DSIAC dataset only.

Table 1: Dataset Breakdown

Dataset Breakdown			
Dataset	**Train**	**Valid**	**Test**
D Total	2800	548	1674
D Ratio 0	92	16	54
D Ratio 1	906	174	540
D Ratio 2	920	160	540
D Ratio 3	882	198	540
Y Total	359138	63017	147043
Y Ratio 0	35914	6320	14686
Y Ratio 1	107742	18945	44073
Y Ratio 2	107742	18868	44149
Y Ratio 3	107740	18883	44135
H Total	76660	29405	68614
H Ratio 0	7666	2890	6912
H Ratio 1	22998	8740	20666
H Ratio 2	22998	8933	20473
H Ratio 3	22998	8842	20563
B Total	143065	7589	17708
B Ratio 0	35766	1873	4451
B Ratio 1	35766	1882	4442
B Ratio 2	35766	1930	4394
B Ratio 3	35766	1904	4420

4.2 Bag-level Classification

Bag-level classification was performed on all four datasets, individually. Each set contained data across all four investigated levels of bag imprecision. Additionally, each level of bag imprecision was

investigated with it's own classifier for set "H". The model used was a ResNet18[58] backbone (consisting of four convolutional blocks) with global average pooling (GAP) and a single fully-connected layer going to a softmax over two outputs. Following best-practices, models were initialized with parameters pre-trained on ImageNet.[59] Each model was trained to optimize Equation 3 for 1000 epochs and parameters were updated with stochastic gradient descent (SGD). Table 2 reports the overall accuracy for the single best model for each dataset (selected from the validation data) on the hold-out test set, as well as the best epoch on the validation set. As can be observed, all models achieved over 93% accuracy for bag-level classification, even on the "expert-designated difficult" datasets. Additionally, the model trained across all levels of bag imprecision for site "H" performed, on average, better than each individual model for a given level of bag imprecision. While counter-intuitive, an explanation for the combined model outperforming alternatives might be that the combination servers as a type of data augmentation, which has been shown to be beneficial to CNN training many times in the literature. Given that each CNN was able to effectively infer bag-level labels, (i.e. use the collection of instances to predict the label of the group), it was assumed that the models contained information about the labels of individual pixels which could be extracted from the networks.

Table 2: Overall test accuracy and best validation epoch for bag-level classification models.

Bag-level Classification Performance		
Dataset	**Accuracy**	**Best Epoch**
D Total	1.000	100
Y Total	0.974	65
H Total	0.950	92
B Total	0.960	69
H Ratio 0	0.943	70
H Ratio 1	0.953	25
H Ratio 2	0.948	40
H Ratio 3	0.930	45

4.3 Comparison of WSSS with Class Activation Maps

Following the bag-level classification experiments, an alternative bag-level classifier was trained for the DSIAC dataset across all levels of bag imprecision. A VGG16 backbone was trained in the same manner outlined in Section 4.2. As with the ResNet backbone, the VGG model achieved 100% bag-level classification performance on the hold-out test set. The VGG model had five convolutional blocks which proceeded pooling and ReLU activations. Each block of convolutional feature maps is referred to as a "stage", where Stage 1 represents early layers presumed to capture general feature information, and where Stage 5 represents late layers deemed to represent class-specific features and localization information. Six class activation map variations (Grad-CAM,[50] Grad-CAM++,[51] LayerCAM,[52] Score-CAM,[60] Ablation-CAM,[54] and Eigen-CAM[55]) were computed in each of the five VGG16 stages for the DSIAC test data. Each CAM was binarized across a range of thresholds in $[0.001, 0.9]$ and compared to the pixel-level groundtruth. Semantic segmentation performance for each method can be observed in Figure 7 and performance in each stage is shown in Figure 8. Table 3 shows the mIoU at a fixed threshold of $\tau = 0.3$, which is consistent with the literature. While overall segmentation performance for each method was poor, there is a clear trend in the results. Specifically, the top scoring methods in late stages were Grad-CAM and Grad-CAM++. This might indicate that localization information in the late layers was very strong, as the model was relying on the large gradient magnitudes (i.e. strong localization) in those layers. In early

Table 3: DSIAC semantic segmentation performance using binarized VGG16 class activation maps at a fixed threshold of $\tau = 0.3$. Results are shown as the mIoU for images predicted as "positive" bags. The best results for each stage are bolded and the second-best are underlined.

DSIAC Semantic Segmentation with Binarized CAMs					
Method	**Stage 1**	**Stage 2**	**Stage 3**	**Stage 4**	**Stage 5**
Grad-CAM[50]	0.084 ± 0.046	0.049 ± 0.052	0.044 ± 0.042	0.156 ± 0.078	$\mathbf{0.274 \pm 0.114}$
Grad-CAM++[51]	0.114 ± 0.103	0.114 ± 0.106	$\underline{0.214 \pm 0.138}$	$\mathbf{0.376 \pm 0.129}$	$\underline{0.242 \pm 0.135}$
LayerCAM[52]	0.078 ± 0.069	$\underline{0.194 \pm 0.140}$	$\mathbf{0.322 \pm 0.168}$	$\underline{0.358 \pm 0.123}$	0.238 ± 0.112
Score-CAM[60]	$\mathbf{0.124 \pm 0.119}$	$\mathbf{0.213 \pm 0.231}$	0.107 ± 0.094	0.113 ± 0.094	0.169 ± 0.120
Ablation-CAM[54]	$\underline{0.115 \pm 0.107}$	0.088 ± 0.083	0.098 ± 0.081	0.168 ± 0.116	0.195 ± 0.093
Eigen-CAM[55]	–	–	0.048 ± 0.042	0.065 ± 0.078	0.045 ± 0.106

stages, however, the effects of diminishing gradients can be seen, as Grad-CAM and Grad-CAM++ have a clear drop-off in performance. Alternatively, LayerCAM, Score-CAM, and Ablation-CAM all showed improved performance in Stages 1-3. LayerCAM utilizes spatial information to improve early layer CAM computation, while Score-CAM and Ablation-CAM substitute backward gradient flow for forward activation passes. These results might suggest a combination of methods taking advantage of both forward activation flow and backward gradient passing would benefit WSSS. Eigen-CAM was not computed in Stages 1 or 2 because of computational resource burden.

4.3.1 Feature Ranking and Selection

Class activation map informed WSSS was further explored by investigating feature selection using CAM pseudo-labels. Following the previous CAM binarization experiments, a LayerCAM model at Stage 4 of the trained VGG16 bag-level classifier was selected to generate pseudo-labels. Two methods for ranking features were explored. First, independent feature ranking was performed for each image by adaptively thresholding activation maps and corresponding LayerCAM pseudo-labels using Otsu's method.[61] The IoU between the binarized feature map and CAM pseudo-label was used as the scoring function. Features were sorted by average IoU performance across all training images. The second ranking was performed by simply sorting the importance values given to each map by Grad-CAM.[50] Figure 10 shows the training mIoU performance for each ordered feature index. As can be observed, the single top performing feature maps obtain, on average, IoU scores of 0.253 (mIoU ranking) and 0.261 (importance ranking). As expected, the feature ranking using mIoU on thresholded activations shows monotonic decreasing performance. Alternatively, the segmentation performance of independent features sorted by Grad-CAM importance shows much more variation. This might suggest that instead of weighting activation maps by their total abilities to cover the target, Grad-CAM may incrementally add activations with less IoU in order to "fill-in" missing areas on target and to reduce redundancy, similar to boosting approaches. Thus, non-linear feature combination may need to be considered to improve CAM computation for WSSS. Figure 11 shows examples of input bags with their top three features selected from ranking with mIoU between activations and CAM pseudo-labels. For the shown example, the first (importance) and second (mIoU) activation feature maps are the same. This is indicative that the feature is important for semantic segmentation. Qualitatively, this ranking is intuitive, as the feature map seems to queue on high response regions of the IR input.

Post feature ranking, the effect of feature set size was explored. A logistic regression classifier was trained K times, where the feature set began with only the fittest feature, and subsequent iterations incrementally added the next fittest feature to the training set. In total, $K = 1472$ classifiers were

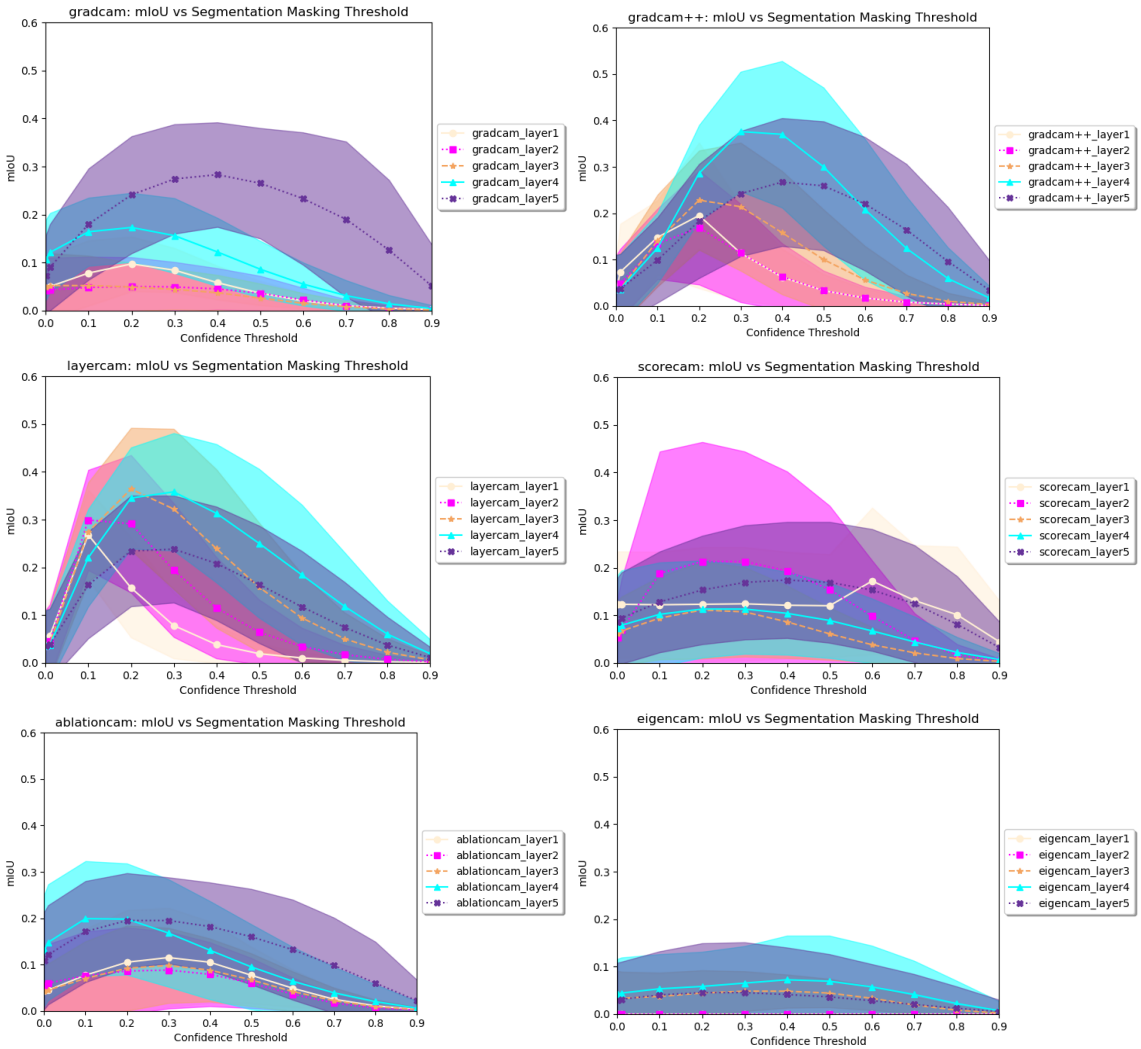

Figure 7: DSIAC CAM segmentation by method. Results show mIoU versus binarization threshold.

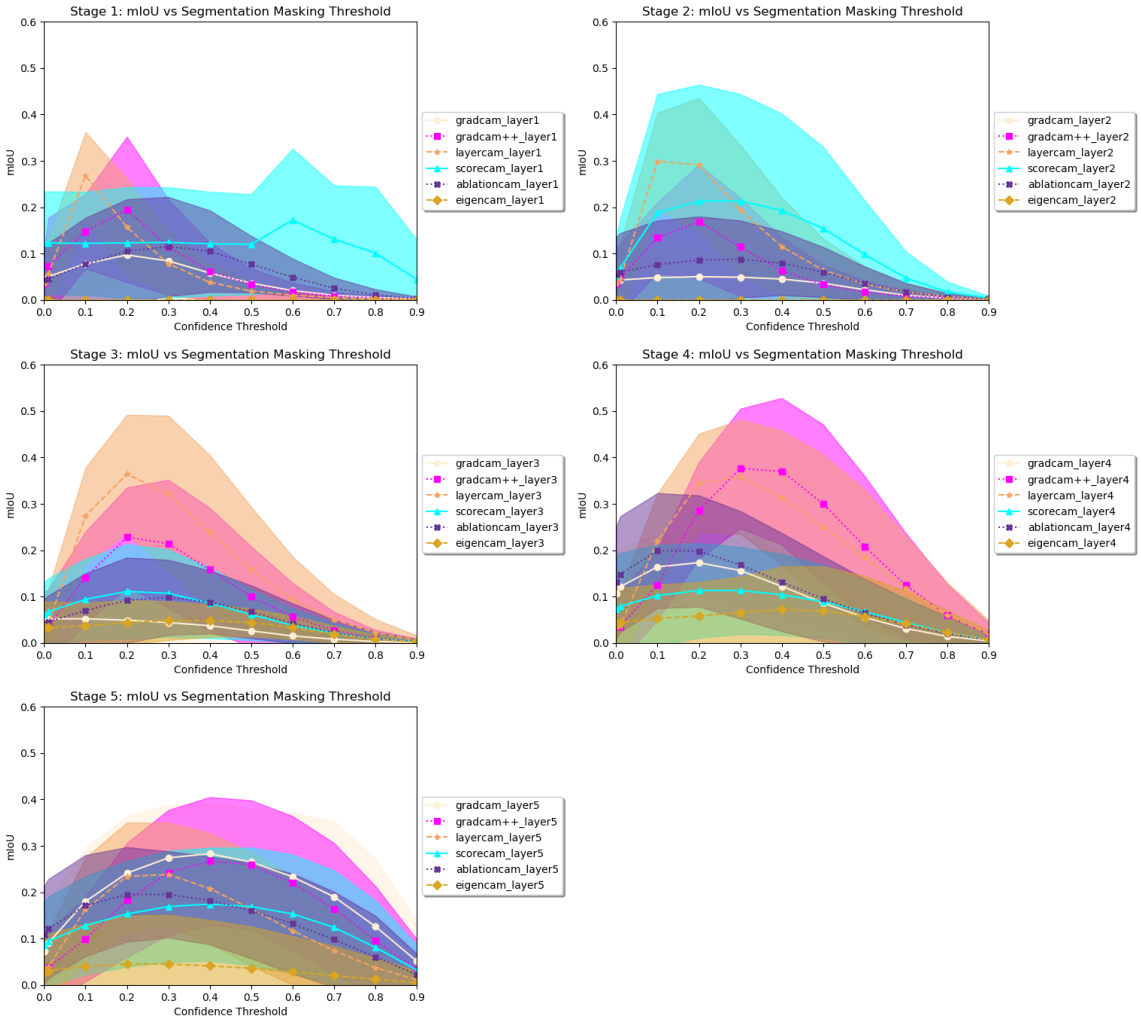

Figure 8: DSIAC CAM segmentation by stage. Results show mIoU versus binarization threshold.

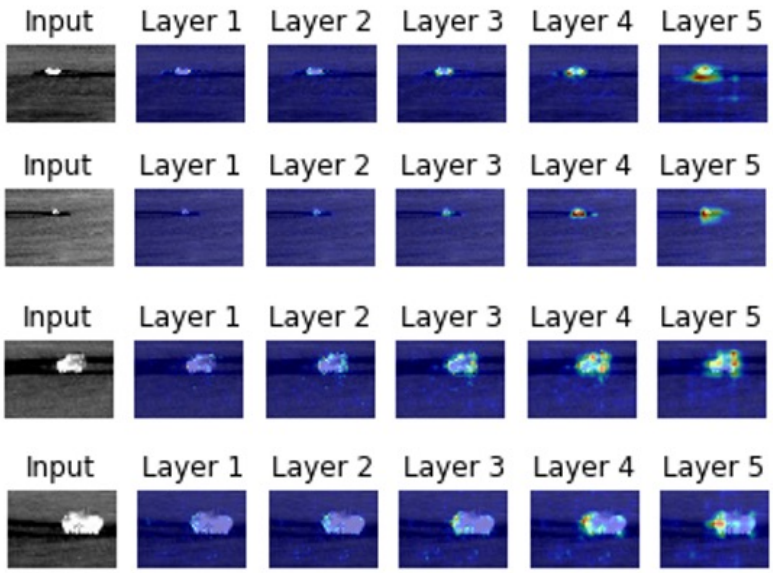

Figure 9: Examples of LayerCAM at various stages of a trained VGG16 network for the "target" class. Layer 1 corresponds to CAMs computed from the initial convolutional block while Layer 5 represents CAMs from the deepest convolutional layers. Earlier layers capture fine-grained information while later layers capture class-specific, localization information.

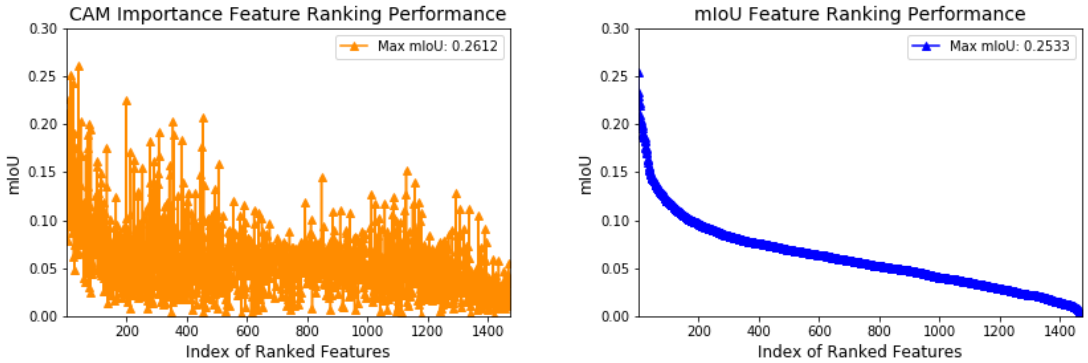

Figure 10: Independent segmentation performance for CAM importance ranked features (left, orange) and mIoU ranked features (right,blue). The single top performing feature maps obtain, on average, IoU scores of 0.253 (mIoU ranking) and 0.261 (importance ranking). The mIoU ranked features show monotonic decreasing segmentation performance while the Grad-CAM ranked features show more variation.

trained to include every feature map from the VGG16 backbone model. The activation maps in $[0, 1]$ were used as input features, while the binary classification labels were taken as the Otsu thresholded LayerCAM outputs for the training bags. Each classifier was tested on the hold-out test set, and the mIoU was computed between the predicted instance-level labels and pixel-level groundtruth. Figure 12 shows the results for each of the two fitness functions. Results are shown only for the first

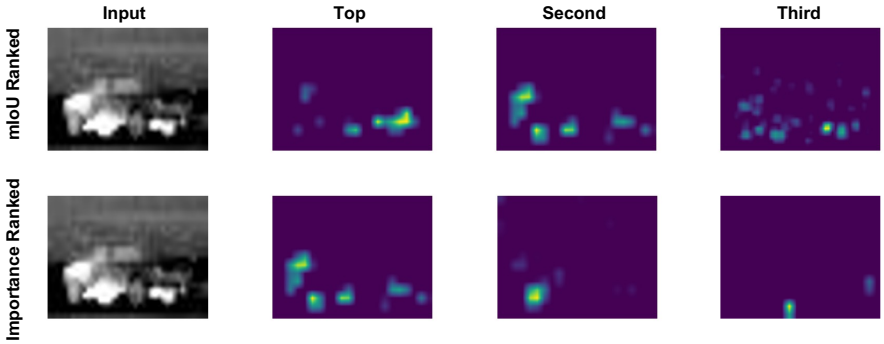

Figure 11: Input images with top three activation maps ranked by mIoU between activations and LayerCAM pseudo-labels (top) and Grad-CAM importance weights (bottom).

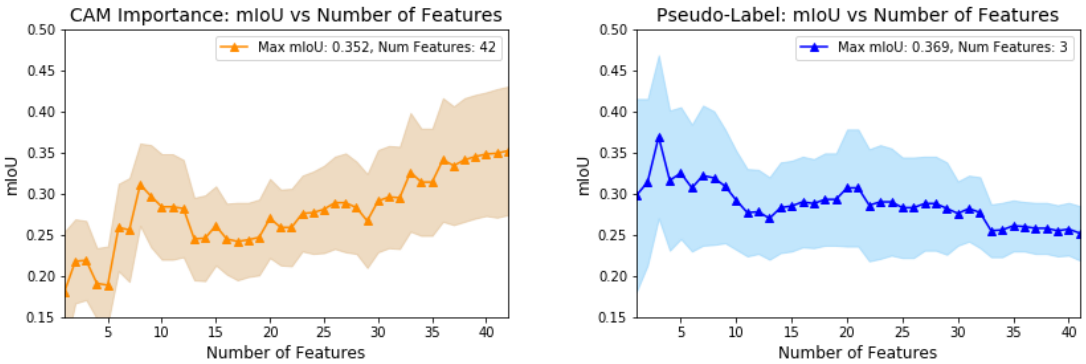

Figure 12: Segmentation performance as a function of number of training features. The solid line shows the mean IoU on the test set and the shaded region shows one standard deviation. The CAM importance fitness (orange) demonstrates a top mIoU of 0.352 at $k = 42$ features. The optimal number of features for the mIoU fitness (blue) is $k = 3$, which gave an average IoU of 0.369.

$k = 42$ feature sets, as performance only declined with additional features. It can be observed that the optimal number of features for the mIoU fitness is $k = 3$, which gave an average IoU of 0.369. The CAM importance fitness gave an optimal mIoU of 0.352 with $k = 42$ features. While the CAM importance fitness performance was on par with the CAM binarization approches for WSSS, the feature selection approach with binarized activations slightly outperformed alternative approaches.

From the results, it can be inferred that selecting feature maps from a trained bag-level classifier may be a viable option for learning instance-level classification features. The implemented feature selection approach benefits from combining both early and late network features. Additionally, feature selection and instance classification are limited by the quality of the CAM pseudo-labels. CAM computation for WSSS could likely be improved if early and late features were fused through intelligent, nonlinear combination.

5. CONCLUSION

This work explored the feasibility of using class activation maps as a mechanism for extracting instance-level classification information from a bag-level inference network. Specifically, six common class activation map approaches in the literature were evaluated on their abilities to perform WSSS by thresholding over a range of confidence values. Pseudo-labels were constructed from a CAM approach, and were used to train an auxiliary instance-level classification model. From experimental results, it was concluded that class activation map approaches can provide adequate localization ability for methods such as seeding, but need further improvement to be utilized for WSSS in infrared imagery, directly. Additionally, it was shown that a set of activation feature maps can be extracted from the bag-level classification network to be used in simpler, auxiliary, instance-level classifiers. Feature selection from CAM pseudo-labels has the ability to combine both early and late features, which can improve WSSS performance in some cases. The methods explored in this paper were baseline approaches to explore the long-investigated problem of instance-level classification under the MIL framework. Both approaches would benefit from improved CAMs. Future work will explore CAM generation which combines features from both early and later stages of a CNN to improve not only target localization, but also instance-level classification.

ACKNOWLEDGMENTS

This work was funded by Army Research Office grant number W911NF-17-1-0213 to support the US Army AFC DEVCOM, C5ISR Center. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies either expressed or implied, by the Army Research Office, Army Research Laboratory, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

REFERENCES

[1] Geng, X., Ji, L., and Zhao, Y., "The basic equation for target detection in remote sensing," (2017).

[2] Chaudhuri, B. and Parui, S., "Target detection: Remote sensing techniques for defence applications," *Defence Science Journal* **45**, 285–291 (04 1995).

[3] Zare, A., Jiao, C., and Glenn, T., "Discriminative multiple instance hyperspectral target characterization," *IEEE Trans. Pattern Anal. Mach. Inteli.* **40**, 2342–2354 (Oct. 2018).

[4] Xu, C., Tao, D., and Rui, Y., "Large-margin weakly supervised dimensionality reduction," *31st International Conference on Machine Learning, ICML 2014* **3**, 2472–2482 (01 2014).

[5] Du, X., *Multiple Instance Choquet Integral For MultiResolution Sensor Fusion*, PhD thesis, Univ. of Missouri, Columbia, MO (Dec. 2017).

[6] Ahn, J. and Kwak, S., "Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation," *CoRR* **abs/1803.10464** (2018).

[7] Zhou, Z.-H., "A brief introduction to weakly supervised learning," *National Science Review* **5**(1), 44–53 (2018).

[8] Li, Y., Guo, L.-Z., and Zhou, Z.-H., "Towards safe weakly supervised learning," *IEEE transactions on pattern analysis and machine intelligence* (2019).

[9] Cinbis, R. G., Verbeek, J., and Schmid, C., "Weakly supervised object localization with multi-fold multiple instance learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **39**(1), 189–203 (2017).

[10] Fu, D. Y., Chen, M. F., Sala, F., Hooper, S. M., Fatahalian, K., and Ré, C., "Fast and three-rious: Speeding up weak supervision with triplet methods," (2020).

[11] Yuan, L., Wen, X., Zhao, L., and Xu, H., "An iterative instance selection based framework for multiple-instance learning," in [*2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI)*], 772–779 (2018).

[12] Wang, P., Zheng, X., Ku, J., and Wang, C., "Multiple-instance learning approach via bayesian extreme learning machine," *IEEE Access* **8**, 62458–62470 (2020).

[13] Carbonneau, M., Cheplygina, V., Granger, E., and Gagnon, G., "Multiple instance learning: A survey of problem characteristics and applications," *CoRR* **abs/1612.03365** (2016).

[14] Rony, J., Belharbi, S., Dolz, J., Ben Ayed, I., McCaffrey, L., and Granger, E., "Deep weakly-supervised learning methods for classification and localization in histology images: a survey," (09 2019).

[15] Dai, J., He, K., and Sun, J., "Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation," *CoRR* **abs/1503.01640** (2015).

[16] Papandreou, G., Chen, L., Murphy, K., and Yuille, A. L., "Weakly- and semi-supervised learning of a DCNN for semantic image segmentation," *CoRR* **abs/1502.02734** (2015).

[17] Boykov, Y. and Jolly, M.-P., "Interactive graph cuts for optimal boundary amp; region segmentation of objects in n-d images," in [*Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*], **1**, 105–112 vol.1 (2001).

[18] Rother, C., Kolmogorov, V., and Blake, A., ""grabcut": Interactive foreground extraction using iterated graph cuts," *ACM Trans. Graph.* **23**, 309–314 (aug 2004).

[19] Li, Y., Sun, J., Tang, C.-K., and Shum, H.-Y., "Lazy snapping," *ACM Trans. Graph.* **23**, 303–308 (aug 2004).

[20] Batra, D., Kowdle, A., Parikh, D., Luo, J., and Chen, T., "icoseg: Interactive co-segmentation with intelligent scribble guidance," in [*2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*], 3169–3176 (2010).

[21] Lin, D., Dai, J., Jia, J., He, K., and Sun, J., "Scribblesup: Scribble-supervised convolutional networks for semantic segmentation," *CoRR* **abs/1604.05144** (2016).

[22] Zhang, J., Zhang, L., Teng, Y., Zhang, X., Wang, S., and Ju, L., "Interactive binary image segmentation with edge preservation," (09 2018).

[23] Bell, S., Upchurch, P., Snavely, N., and Bala, K., "Material recognition in the wild with the materials in context database," *CoRR* **abs/1412.0623** (2014).

[24] Bearman, A., Russakovsky, O., Ferrari, V., and Fei-Fei, L., "What's the point: Semantic segmentation with point supervision," in [*Computer Vision – ECCV 2016*], Leibe, B., Matas, J., Sebe, N., and Welling, M., eds., 549–565, Springer International Publishing, Cham (2016).

[25] Li, Z., Chen, Q., and Koltun, V., "Interactive image segmentation with latent diversity," in [*2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*], 577–585 (2018).

[26] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L., "Microsoft coco: Common objects in context," in [*Computer Vision – ECCV 2014*], Fleet, D., Pajdla, T., Schiele, B., and Tuytelaars, T., eds., 740–755, Springer International Publishing, Cham (2014).

[27] Zhou, B., Khosla, A., A., L., Oliva, A., and Torralba, A., "Learning Deep Features for Discriminative Localization.," *CVPR* (2016).

[28] Dietterich, T. G., Lathrop, R. H., and Lozano-Pérez, T., "Solving the multiple instance problem with axis-parallel rectangles," *Artificial Intelligence* **89**(1), 31 – 71 (1997).

[29] Ilse, M., Tomczak, J. M., and Welling, M., "Attention-based deep multiple instance learning," *CoRR* **abs/1802.04712** (2018).

[30] Du, X. and Zare, A., "Multiple instance choquet integral classifier fusion and regression for remote sensing applications," *IEEE Transactions on Geoscience and Remote Sensing* **57**, 2741–2753 (May 2019).

[31] Du, X. and Zare, A., "Multi-resolution multi-modal sensor fusion for remote sensing data with label uncertainty," *CoRR* **abs/1805.00930** (2018).

[32] Zare, A., Cook, M., Alvey, B., and Ho, D. K., "Multiple instance dictionary learning for subsurface object detection using handheld emi," in [*Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XX, 94540G*], *Proc. SPIE* (May 2015).

[33] McCurley, C. H., Bocinsky, J., and Zare, A., "Comparison of hand-held wemi target detection algorithms," in [*Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XXIV, 110120U*], *Proc.SPIE* **11012** (May 2019).

[34] Meerdink, S., Bocinsky, J., Zare, A., Kroeger, N., McCurley, C. H., Shats, D., and Gader, P., "Multi-target multiple instance learning for hyperspectral target detection," *IEEE Transaction on Geoscience and Remote Sensing (TGRS)* (2021).

[35] Zhou, B., Khosla, A., Lapedriza, À., Oliva, A., and Torralba, A., "Learning deep features for discriminative localization," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* , 2921–2929 (2016).

[36] Yu, G., Zare, A., Xu, W., Matamala, R., Reyes-Cabrera, J., Fritschi, F. B., and Juenger, T. E., "Weakly supervised minirhizotron image segmentation with mil-cam," *16th European Conference on Computer Vision (ECCV) Workshop on Computer Vision Problems in Plant Phenotyping (CVPPP 2020)* (In Press).

[37] Zhou, B., Sun, Y., Bau, D., and Torralba, A., "Interpretable basis decomposition for visual explanation," in [*Proceedings of the European Conference on Computer Vision (ECCV)*], (September 2018).

[38] Jalwana, M. A. A. K., Akhtar, N., Bennamoun, M., and Mian, A., "CAMERAS: enhanced resolution and sanity preserving class activation mapping for image saliency," *CoRR* **abs/2106.10649** (2021).

[39] Jung, H. and Oh, Y., "LIFT-CAM: towards better explanations for class activation mapping," *CoRR* **abs/2102.05228** (2021).

[40] Lee, K. H., Park, C., Oh, J., and Kwak, N., "LFI-CAM: learning feature importance for better visual explanation," *CoRR* **abs/2105.00937** (2021).

[41] Zhang, A., Wang, X., Fang, C., Shi, J., Chua, T., and Chen, Z., "A-FMI: learning attributions from deep networks via feature map importance," *CoRR* **abs/2104.05527** (2021).

[42] Wei, Y., Xiao, H., Shi, H., Jie, Z., Feng, J., and Huang, T. S., "Revisiting dilated convolution: A simple approach for weakly- and semi-supervised semantic segmentation," in [*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*], (June 2018).

[43] Laradji, I. H., Vázquez, D., and Schmidt, M., "Where are the masks: Instance segmentation with image-level supervision," *CoRR* **abs/1907.01430** (2019).

[44] Kolesnikov, A. and Lampert, C. H., "Seed, expand and constrain: Three principles for weakly-supervised image segmentation," *CoRR* **abs/1603.06098** (2016).
[45] Hong, S., Oh, J., Han, B., and Lee, H., "Learning transferrable knowledge for semantic segmentation with deep convolutional neural network," *CoRR* **abs/1512.07928** (2015).
[46] Li, K., Wu, Z., Peng, K., Ernst, J., and Fu, Y., "Tell me where to look: Guided attention inference network," *CoRR* **abs/1802.10171** (2018).
[47] Shi, Z., Yang, Y., Hospedales, T., and Xiang, T., "Weakly supervised learning of objects, attributes and their associations," (09 2014).
[48] Zhu, Y., Zhou, Y., Xu, H., Ye, Q., Doermann, D., and Jiao, J., "Learning instance activation maps for weakly supervised instance segmentation," in [*2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*], 3111–3120 (2019).
[49] Li, X., Ma, H., and Luo, X., "Weaklier supervised semantic segmentation with only one image level annotation per category," *IEEE Transactions on Image Processing* **29**, 128–141 (2020).
[50] Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D., "Grad-cam: Visual explanations from deep networks via gradient-based localization," in [*2017 IEEE International Conference on Computer Vision (ICCV)*], 618–626 (2017).
[51] Chattopadhay, A., Sarkar, A., Howlader, P., and Balasubramanian, V. N., "Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks," in [*2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*], 839–847 (2018).
[52] "Layercam: Exploring hierarchical class activation maps for localization," *IEEE Transactions on Image Processing* **30**, 5875–5888 (2021).
[53] Wang, H., Du, M., Yang, F., and Zhang, Z., "Score-cam: Improved visual explanations via score-weighted class activation mapping," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* , 2921–2929 (2020).
[54] Desai, S. and Ramaswamy, H. G., "Ablation-cam: Visual explanations for deep convolutional network via gradient-free localization," in [*2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*], 972–980 (2020).
[55] Muhammad, M. B. and Yeasin, M., "Eigen-cam: Class activation map using principal components," *CoRR* **abs/2008.00299** (2020).
[56] Latham, A. C., *Multiple-Instance Feature Ranking*, Master's thesis, Case Western Reserve University, Cleveland, OH (August 2015).
[57] DSIAC, D. S. I. A. C., "DSIAC MS-003-DB Algorithm Development Database." https://www.dsiac.org/resources/research-materials/cds-dvds-databases-digital-files/atr-algorithm-development-image (2014).
[58] He, K., Zhang, X., Ren, S., and Sun, J., "Deep residual learning for image recognition," *CoRR* **abs/1512.03385** (2015).
[59] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L., "Imagenet: A large-scale hierarchical image database," in [*2009 IEEE conference on computer vision and pattern recognition*], 248–255, Ieee (2009).
[60] Wang, H., Du, M., Yang, F., and Zhang, Z., "Score-cam: Improved visual explanations via score-weighted class activation mapping," *CoRR* **abs/1910.01279** (2019).
[61] Otsu, N., "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics* **9**(1), 62–66 (1979).