Multi-Label Visual Feature Learning with Attentional Aggregation

Ziqiao Guan
Stony Brook University
ziguan@cs.stonybrook.edu

Kevin G. Yager
Brookhaven National Laboratory
kyager@bnl.gov

Dantong Yu
New Jersey Institute of Technology
dtyu@njit.edu

Hong Qin
Stony Brook University
qin@cs.stonybrook.edu

Abstract

Today convolutional neural networks (CNNs) have reached out to specialized applications in science communities that otherwise would not be adequately tackled. In this paper, we systematically study a multi-label annotation problem of x-ray scattering images in material science. For this application, we tackle an open challenge with training CNNs — identifying weak scattered patterns with diffuse background interference, which is common in scientific imaging. We articulate an Attentional Aggregation Module (AAM) to enhance feature representations. First, we reweight and highlight important features in the images using data-driven attention maps. We decompose the attention maps into channel and spatial attention components. In the spatial attention component, we design a mechanism to generate multiple spatial attention maps tailored for diversified multi-label learning. Then, we condense the enhanced local features into non-local representations by performing feature aggregation. Both attention and aggregation are designed as network layers with learnable parameters so that CNN training remains fluidly end-to-end, and we apply it in-network a few times so that the feature enhancement is multi-scale. We conduct extensive experiments on CNN training and testing, as well as transfer learning, and empirical studies confirm that our method enhances the discriminative power of visual features of scientific imaging.

1. Introduction

In recent years, deep learning and convolutional neural networks (CNNs) have moved on from success in general computer vision problems and applications, to working with more specialized data and problems, *e.g*. applications in science communities. These dedicated applications usually come with a relatively small dataset and unique challenges with data distributions. This causes a lot of trouble for CNNs to capture them properly and realize their full potential. From a feature space's point of view, visual features in scientific imaging lie in a low-dimensional, highly restricted and "narrow" subspace compared to natural images, and the feature representations must be enhanced properly to work with CNNs.

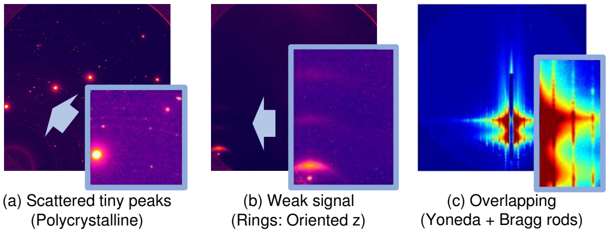

(a) Scattered tiny peaks (Polycrystalline)
(b) Weak signal (Rings: Oriented z)
(c) Overlapping (Yoneda + Bragg rods)

Figure 1. Examples of x-ray scattering images. The original image is a single-channel intensity map captured on the detector; it is shown here with false color for visualization purposes. In the enlarged window of each image, we highlight the **challenging features**, and **the attributes that should be deduced from the said features**: (a) related peaks are scattered and far apart (Polycrystalline); (b) signal too dim (Rings: Oriented z); (c) thin vertical bars (Bragg rods) overlaying bright regions (Yoneda). Enlarged windows of (a) and (b) are brightened to show weak signals.

We will illustrate this difficulty of feature space with examples of x-ray scattering images, which are the subject of our analysis in this paper. In x-ray scattering, a beam of x-rays is directed through a material sample of interest, diffracted by the ordering within the sample; the far-field pattern of scattered rays is captured on an x-ray area detector. This diffractive imaging process is essentially described by a Fourier transform of the sample's real space distribution $\rho(\mathbf{r})$:

$$I(\mathbf{q}) = \left| \int_V \rho(\mathbf{r}) \exp(i\mathbf{q} \cdot \mathbf{r}) \, \mathrm{d}V \right|^2, \qquad (1)$$

from which only the intensity is captured, while the phase is lost. Thus, the diffraction is non-invertible. To circumvent the inverse reconstruction problem, material scientists can directly inspect the image to deduce a set of characteristics, *e.g.* visual appearance ('halo' or 'ring'), style variations ('isotropic' or '6-fold symmetric'), material type ('powder' or 'polycrystalline') or crystal lattice structure ('BCC' or 'lamellar'). In other words, this is an *multi-label* image annotation problem in the reciprocal space (or q-space); multi-label meaning the aforementioned attributes are non-mutually-exclusive.

When we apply CNNs to this multi-label annotation problem, the key challenge comes from the weak and scattered patterns in the x-ray scattering images. Here we show a few difficult cases in Figure 1. In Figure 1 (a), a sample is bearing the attribute "polycrystalline", implied by the set of high-intensity peaks in the image. The peaks themselves are local, yet the "polycrystalline" character can only be inferred by identifying a non-local set of peaks combined. If a CNN unit is to perceive a window large enough to capture all the peaks, it will take in a much bigger portion of background and possibly other overlapping signals, overwhelming the peaks in question. Figure 1 (b)(c) shows two other cases of difficult attributes caused by weak visual features and overlapping. For weak, noisy features like these, we need effective measures to encode them robustly.

In this paper, we present the Attentional Aggregation Module (AAM), a modularized two-step strategy to enhance feature representations. Given a certain set of convolutional feature maps, first we attempt to explicitly reweight and highlight key features with attention mechanism. When generating attention maps, We decompose them into channel and spatial attention components for better separation between them. In the spatial attention component, we design a mechanism to generate multiple spatial attention maps to apply to partitions of the feature maps, which diversifies the attentional features for different attributes of interest. Then, we attempt to condense those scattered sparse features by feature aggregation. We extend the classic Bag-of-Words (BoW) with learnable parameters so that it can be performed in-network. We apply AAM multiple times in the network at different depths so that the feature enhancement is multi-scale.

Our main contributions in this paper are as follows:

- We designed the Attentional Aggregation Module using differentiable layers and learnable parameters, which enabled end-to-end forward and backward flows in the CNN, and repeated deployment in multiple CNN layers. Thus, feature enhancement is seamless and multi-scale;
- We improved the attention modules with new tactics and multiple spatial attention maps to specifically tackle the multi-label learning problem, and demonstrated their benefits via experiments.

2. Related Work

X-ray scattering image analysis. Studying x-ray scattering imagery is an interdisciplinary effort of computer vision and scattering communities. There are unsupervised methods such as spectral clustering [41] and diffusion-based clustering [10], as well as supervised methods such as [13] using handcrafted image descriptors. CNN based techniques are first used in other similar scientific dataset problems, *e.g.* [40] applies a CNN to classify x-ray protein crystallization images. For the x-ray scattering image annotation problem, [21] performs 1D convolutions on the circular average curve of the images; [32] implements residual learning [6] and convolutional autoencoders; [5] proposes a joint learning framework with physics-aware feature transform [34]. Despite the general success of these methods, some attributes are connected to more intricate features as we explained in Figure 1, and thus they are hard to observe by even humans and so remain difficult for machine learning methods.

Feature aggregation. Typical feature aggregation methods organize generic features by encoding statistics of a collection of features, *e.g.*, Bag-of-Words (BoW) [29], VLAD [12, 2], Fisher Vector [23, 24], spatial pyramid matching [15] and Bag-of-Feature-Graphs [8].

Deep CNN generates dense collections of features. Many works attempt to incorporate classic feature aggregation methods, *e.g.* MOP-CNN [4] pools VLADs of multi-scale CNN activations, and NetVLAD [1] presents a learnable VLAD in deep CNNs. Later [19] generalized the learnable construction of NetVLAD to BoW and Fisher Vector. On the other hand, since multi-scale is naturally implied in the depths of CNNs, researchers have tried countless network designs to fuse cross-layer features as a form of feature aggregation, *e.g.* U-Net [25] reusing mid-layer feature maps. More sophisticated connection designs include recombinator networks [7] and stacked hourglass networks [20], and [17] has made some detailed discussions about various pathway designs in multi-scale analysis. However, many of these methods rely on reasonably good local features to aggregate. They mostly work like representing the composition of a scene given all the objects have been well depicted. As for x-ray scattering images, the features are tricky to capture even at the local level and we need explicit strategies to boost local features so that aggregation can be effective.

Attention mechanism. It is known that humans perceive images not by observing the entire scene, but focusing their attention on salient regions [14]. In computer vision, researchers have attempted to mimic attention for feature learning [14] and generative models [31, 37]. For computing attention maps, researchers have proposed to use fully-connected MLP [37], convolutions and residual convolutions [33], and correlations to encode non-local interactions [42]. With the recent success of channel attention in SENet [9, 43], dual attention, which is a decomposition of channel and spatial attention, has become a popular method to model attention [3, 18, 35].

3. Attentional Aggregation Module

In this section we describe the Attentional Aggregation Module (AAM), which is designed as learnable network layers and applied several times to enhance the features in a multi-scale fashion. The structure of AAM is described in Algorithm 1, and the overall network is shown in Figure 2.

3.1. Attention: Local Refinement

Given a feature map $\mathbf{F} \in \mathbb{R}^{C\times H\times W}$, we first attempt to refine the features locally using attention mechanism. Attention in features is essentially a reweighting process to highlight certain parts of the features, represented by multiplying $\mathbf{F}$ with attention map $\mathbf{M}$. When the attention module is trained to compute from the data $\mathbf{F}$ to mimic human attention, it can be written as

$$\mathbf{F}' = \mathbf{M}(\mathbf{F}) \otimes \mathbf{F}, \tag{2}$$

where $\otimes$ represents elementwise multiplication, and $\mathbf{M}(\cdot)$ becomes a data-driven attention estimator.

Dual attention. We use a dual attention to approximate $\mathbf{M}$, *i.e.* decompose the overall attention into 2 multiplicative components: 1D channel attention and 2D spatial attention

$$\mathbf{F}' = \mathbf{M_C}(\mathbf{F}) \otimes \mathbf{F}, \quad \mathbf{F}'' = \mathbf{M_S}(\mathbf{F}') \otimes \mathbf{F}'. \tag{3}$$

We can think of $\mathbf{M_C}$ as modulating the C feature channels, or amplifying/suppressing the C feature detectors if we consider each channel as a specific detector; and $\mathbf{M_S}$ focuses on pixel locations. This is similar to a low-rank matrix decomposition and the attention in different dimensions can be better separated.

We follow a typical dual attention model — Convolutional Block Attention Module [35] (CBAM, shown in Figure 3) — to formulate $\mathbf{M_C}$ and $\mathbf{M_S}$. The channel attention component first spatially pools $\mathbf{F}$ to a C-dimension vector, and then encodes the vector with a 2-layer fully-connected network (or Multi-Layer Perceptron, MLP):

$$\mathbf{M_C}(\cdot) = \sigma \circ \mathrm{MLP} \circ \mathrm{Pool}_\mathbf{C}(\cdot), \tag{4}$$

where σ is sigmoid. Similarly, the spatial attention component first performs a channelwise pooling to generate a $H \times W$ matrix, and then computes a convolution:

$$\mathbf{M_S}(\cdot) = \sigma \circ \mathrm{Conv} \circ \mathrm{Pool}_\mathbf{S}(\cdot). \tag{5}$$

Finally, we add up the reweighted $\mathbf{F}''$ with $\mathbf{F}$ and normalize it with batch normalization [11] (BN) to prevent feature degradation due to successive multiplications with values between $[0, 1]$ [33]:

$$\mathbf{F}_{\mathrm{attn}} = \mathrm{BN}(\mathbf{F} + \mathbf{F}''). \tag{6}$$

Unfortunately, the original CBAM in the network does not improve annotation on our x-ray scattering datasets, because smaller dataset size and fewer positive samples cause more difficulty to learn. We made a few improvements for the attention mechanism, as follows:

Pre-activation normalization of attention maps. We find that both channel and spatial attention components tend to saturate sigmoid and generate attention maps that are all 0 or 1. To correct this, we add a BN prior to sigmoid, in both channel and spatial components, to stabilize the range of features. The channel attention component is now written as

$$\mathbf{M_C}(\cdot) = \sigma \circ \mathrm{BN} \circ \mathrm{MLP} \circ \mathrm{Pool}_\mathbf{C}(\cdot). \tag{7}$$

With normalization, the generated attention maps can actually have values in $[0, 1]$.

Multiple spatial attention maps. We argue that for spatial attention, one single spatial map to reweight all C feature channels does not account for the different features that these channels specialize in for multi-label annotation. Instead, we feed the feature $\mathbf{F}'$ into p duplicate branches of the spatial attention component:

$$\{\mathbf{M}^i_\mathbf{S} = \sigma \circ \mathrm{BN}^i \circ \mathrm{Conv}^i \circ \mathrm{Pool}_\mathbf{S} \mid 1 \leq i \leq p\}. \tag{8}$$

Then, we split $\mathbf{F}'$ into p uniform slices along the channel dimension C, and reweight each C/p-channel slice $\mathbf{F}'_i$ with $\mathbf{M}^i_\mathbf{S}(\mathbf{F}')$. Then we concatenate all the slices into a p-way reweighted $\mathbf{F}''$. In our experiments, we set $p = 4$.

Specialized loss for multi-maps. We further push the p spatial attention branches to diversify. For this purpose, we partition the attributes into p groups and associate each group with one spatial attention branch. For each branch during training, we compute the label loss with respect to its own attribute group, and update its parameters with this specialized loss/gradient only.

For instance, we denote the image attributes as $\mathbf{Y} = \{y_1, y_2, \ldots, y_n\}$. We may associate $\mathbf{Y}^{(1)} = \{y_1, \ldots, y_{n/p}\}$ with $\mathbf{M}^1_\mathbf{S}(\cdot)$, $\mathbf{Y}^{(2)} = \{y_{n/p+1}, \ldots, y_{2n/p}\}$ with $\mathbf{M}^2_\mathbf{S}(\cdot)$, and so forth. Pick a label loss function, *e.g.* binary cross entropy:

$$\mathcal{L}^\Lambda_{\mathrm{BCE}} = -\frac{1}{|\Lambda|}\sum_j^\Lambda y_j \log o_j + (1 - y_j)\log(1 - o_j), \tag{9}$$

where Λ is a certain attribute set, y_j is the true value of attribute j, and o_j is the prediction value. For training the rest of the network, our objective is to fit all the attribute predictions, $\Lambda = \mathbf{Y}$; For $\mathbf{M}^i_\mathbf{S}(\cdot)$, $\Lambda = \mathbf{Y}^{(i)}$.

The full attention module is shown in Figure 4.

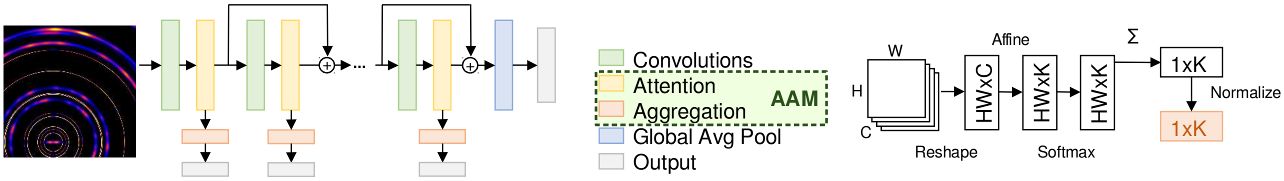


Figure 2. Left: Architecture of our network with Attentional Aggregation Modules. Right: Structure of the aggregation module.

```
ALGORITHM 1: Attentional Aggregation Module.
  Input: Convolutional feature $\mathbf{F} \in \mathbb{R}^{C\times H\times W}$.
  Output: Attentional feature $\mathbf{F}_{\text{attn}} \in \mathbb{R}^{C\times H\times W}$ and
          aggregated vector $\mathbf{f}_{\text{aggr}} \in \mathbb{R}^{K}$.
  Compute channelwise attention $\mathbf{M_C}(\mathbf{F}), \mathbf{F}'$ using (3)(7);
  Compute spatial attention maps $\{\mathbf{M}^i_\mathbf{S}(\mathbf{F}')\}_i$ using (8);
  Split $\mathbf{F}'$ along the channel dimension into $p$ slices $\{\mathbf{F}'_i\}_i$;
  Compute spatial attention
    $\mathbf{M_S}(\mathbf{F}') = \text{Concat}(\{\mathbf{M}^i_\mathbf{S}(\mathbf{F}') \otimes \mathbf{F}'_i\}_i)$, and $\mathbf{F}''$ using (3);
  Compute attentional features $\mathbf{F}_{\text{attn}}$ using (6);
  Reshape $\mathbf{F}_{\text{attn}}$ to $\bar{\mathbf{F}}_{\text{attn}} \in \mathbb{R}^{HW\times C}$;
  Compute $\mathbf{A} = \bar{\mathbf{a}}(\bar{\mathbf{F}}_{\text{attn}}) \in \mathbb{R}^{HW\times K}$ using (11);
  Sum $\mathbf{A}$ up along the columns and normalize to unit 2-norm
    to get $\mathbf{f}_{\text{aggr}}$;
```

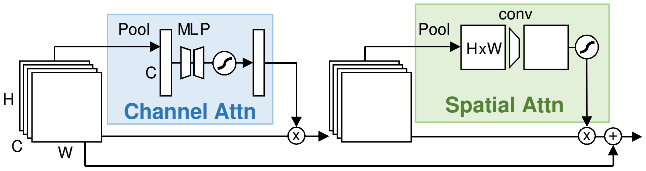


Figure 3. Structure of CBAM [35].

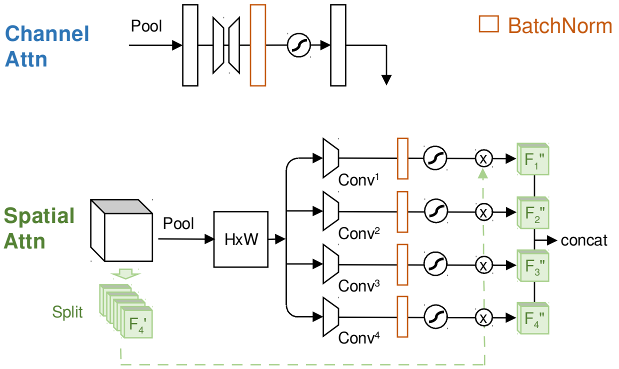


Figure 4. Our channel and spatial attention modules.

3.2. Aggregation: Non-local Representation

We then aggregate the local attention feature $\mathbf{F}_{\text{attn}}$ to condense non-local feature representations. Our feature aggregation module is based on the classic Bag-of-Words (BoW). Formally, given a corpus of sampled feature vectors (words), BoW computes a clustering to determine a set of K clusters $\{C_k\}$ and their centroids $\{\mathbf{c}_k\}$. For an image with N extracted features $\{\mathbf{f}_i\}$, its BoW is its feature distribution with respect to the clusters

$$\left[\sum_{i=1}^{N} a_1(\mathbf{f}_i), \sum_{i=1}^{N} a_2(\mathbf{f}_i), \ldots, \sum_{i=1}^{N} a_K(\mathbf{f}_i)\right], \qquad (10)$$

where $a_k(\cdot)$ is an assignment function, typically a hard 0-1 assignment determined by the nearest $\mathbf{c}_k$.

Consider $\mathbf{F}_{\text{attn}}$ as an $H \times W$ pool of C-dimensional feature words and we attempt to aggregate them spatially. In order to perform aggregation in-network and have it compatible with back-propagation, the key is a differentiable formulation for the cluster assignment $a_k(\cdot)$. NetVLAD [1] manages to do this by replacing the hard assignment $a_k(\cdot)$ with soft assignment and relaxing the learnable parameters. Learnable BoW is similar, as we need to replace $a_k(\cdot)$ with a softmax

$$\bar{\mathbf{a}}(\mathbf{x}) = \text{softmax}(\{\mathbf{w}_k^T\mathbf{x} + b_k\}_{1\le k\le K}), \qquad (11)$$

and then all the K-assignment vectors are summed up and normalized, denoted as $\mathbf{f}_{\text{aggr}}$. In our experiments, we set $K = 64$. The aggregation module is shown in Figure 2 on the right.

3.3. Network Architecture

We adopt VGG-16 [28] as our backbone network. VGG-16 is naturally divided into 5 blocks with pooling layers in between, implying different scales. We put in residual bypass [6] over each block to separate the features at each scale. After the last convolution, we feed the feature maps into a global average pooling layer and a sigmoid output.

We plug in an AAM before the addition in each of the aforementioned residual bypasses. The AAM computes $\mathbf{F}_{\text{attn}}$ and $\mathbf{f}_{\text{aggr}}$. We pass $\mathbf{F}_{\text{attn}}$ through onto subsequent CNN layers, and $\mathbf{f}_{\text{aggr}}$ leads to a fully-connected layer and a sigmoid, which serves as a side output $\mathbf{y}_s, 1 \le s \le 5$. We fit the side output to ground truth attributes, similar to [30]. The purpose is to stimulate multi-scale features to better relate to attributes. As a result, the overall loss function is:

$$\mathcal{L} = \mathcal{L}^{\Lambda}_{\text{BCE}}(\mathbf{y}, \mathbf{o}) + w\sum_{s=1}^{5} \mathcal{L}^{\Lambda}_{\text{BCE}}(\mathbf{y}_s, \mathbf{o}), \qquad (12)$$

where Λ is a specialized attribute set (described in Section 3.1), $\mathbf{y}$ is the output from the last CNN layer, $\mathbf{y}_s$ is a side output, and $\mathbf{o}$ is the real attribute. We set $w = 0.2$.

		mAP
(a)	VGG-16	0.6760
(b)	+Aggregation	0.7224
(c)	+Normalized CBAM	0.7398
(d)	+Residual Attention	0.7248
(e)	+SENet	0.7304
(f)	+AAM−specialized loss	0.7407
(g)	+AAM	**0.7433**

Table 1. mAPs with different network setups and other attention mechanisms, on synthetic dataset.

	mAP	
Method	Single	Mixed
VGG-16	0.8312	0.7997
ResNet-50	0.8231	0.7084
DVFB-CNN	0.8513	0.7989
SENet	0.8723	0.8071
Residual Attention	**0.8837**	0.8183
Ours	0.8739	**0.8225**

Table 2. Comparison with state-of-the-art deep learning methods, on experimental dataset.

4. Experiments

4.1. Datasets and Metrics

We use the following 3 datasets to evaluate our network:

Synthetic Dataset. We use simulation software [38] to generate high volumes of simulated x-ray scattering images with auto-generated attributes. The software models x-ray imagery with high fidelity [22, 39, 27] and adapts well for machine learning models that extend to real data [32, 5]. For comparisons with previously reported methods, we generate 45,000 images for training and 5,000 images for testing as our synthetic dataset. We pick 20 attributes with typical visual appearances and/or physical meanings to predict.

Experimental Dataset. We take the experimental dataset assembled in [5] to assess our method with real experimental data. The experimental dataset is collected from various x-ray beamline facilities and fully annotated by a domain expert. It is organized into 2 groups: "single" consists of different image captures with homogeneous experiment setups (beam center position, detector placement etc.), and "mixed" where experiment setups are diverse. The single dataset has 2,000 training images, 429 testing images and 12 attributes; and the mixed dataset has 2,300 training images, 418 testing images and 20 attributes. These attributes are not the same as those in the synthetic dataset.

Fashion-MNIST. To test the AAM with other forms of data, we choose Fashion-MNIST [36] as a general purpose dataset. It is designed as a drop-in replacement of the heavily-used MNIST [16] and consists of grayscale images of 10 types of clothing articles of size 28×28, 60,000 training samples and 10,000 testing samples.

For synthetic and experimental datasets, the annotation is multi-label, and we report the average precisions (APs) per attribute and mean average precision (mAP); Fashion-MNIST is a multi-class classification dataset, and we report the classification accuracy.

4.2. Ablation Studies

To verify the effect of the attention and aggregation modules, we trained and tested the CNN on synthetic dataset, with our proposed strategies added incrementally. They are listed in Table 1 as: (a) barebone VGG-16, (b) with aggregation module, (c) with aggregation and normalized CBAM, (f) with AAM, but without specialized loss, and (g) with AAM. Since CBAM has the saturation problem (described in Section 3.1), we added a BN in (c).

We trained all the 5 networks end-to-end and directly evaluated the outputs. For specialized loss, the 20 attributes were grouped as follows: (1) major visual elements: *Diff low-q*, *Diff hi-q*, *Halo*, *Higher ord*, and *Ring*; (2) symmetry: *Sym halo*, *Sym ring*, *2-fold*, *4-fold*, and *6-fold*; (3) texture: *Anisotropic*, *Isotropic*, *Spotted*, and *Textured*; (4) visual style: *Orientation: sharp*, *Orientation: broad*, *Orientation: interm*, *Width: sharp*, *Width: broad* and *Width: interm*. We report the APs per attribute in Figure 5, and the mAPs in Table 1.

We can see consistent improvements from (a)(b)(c)(f)(g) with the added components:

- The performance gain from (a) to (b) comes from feature aggregation and early side supervisions, which effectively shape the multi-scale features as proven by GoogLeNet [30];
- (c)(f)(g) shows the benefit of multiple spatial attention maps, and demonstrates that specialized loss provides the additional information to effectively train more learnable parameters and operations.

4.3. Comparison with Other Attention Methods

We also compared AAM with some other attention mechanisms: (d) SENet [9] and (e) residual attention module [33], as shown in Table 1. We kept the VGG-16 with AAM structure unchanged but swapped all the attention modules with the other methods.

The results show that AAM enables better precision. Essentially, SENet is channel attention only, and residual attention does not decompose the attention into channel and spatial components, while AAM exploits both dual attention and residual attention, and also utilizes multiple spatial attention maps.

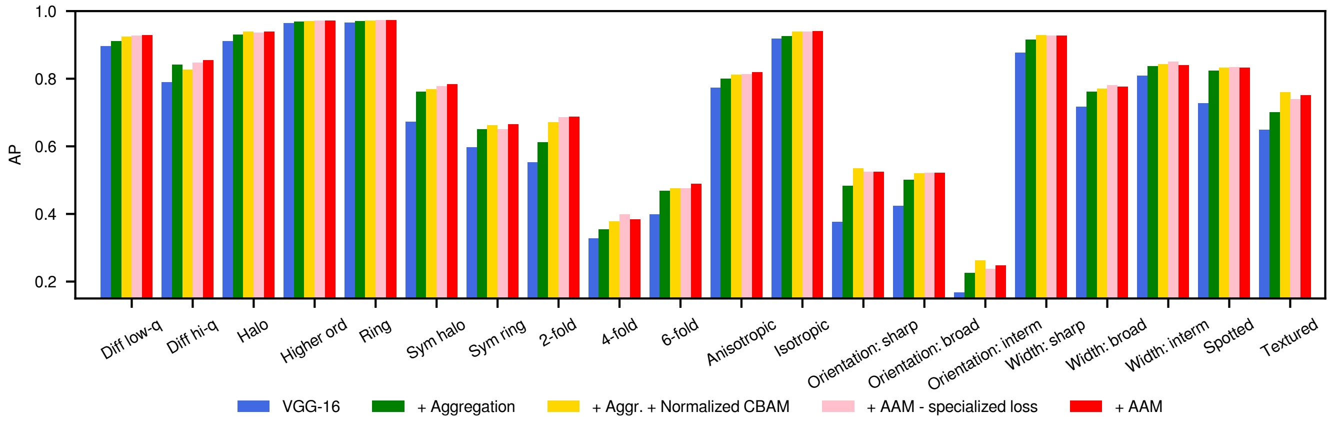


Figure 5. APs per category with different network setups, on synthetic dataset.

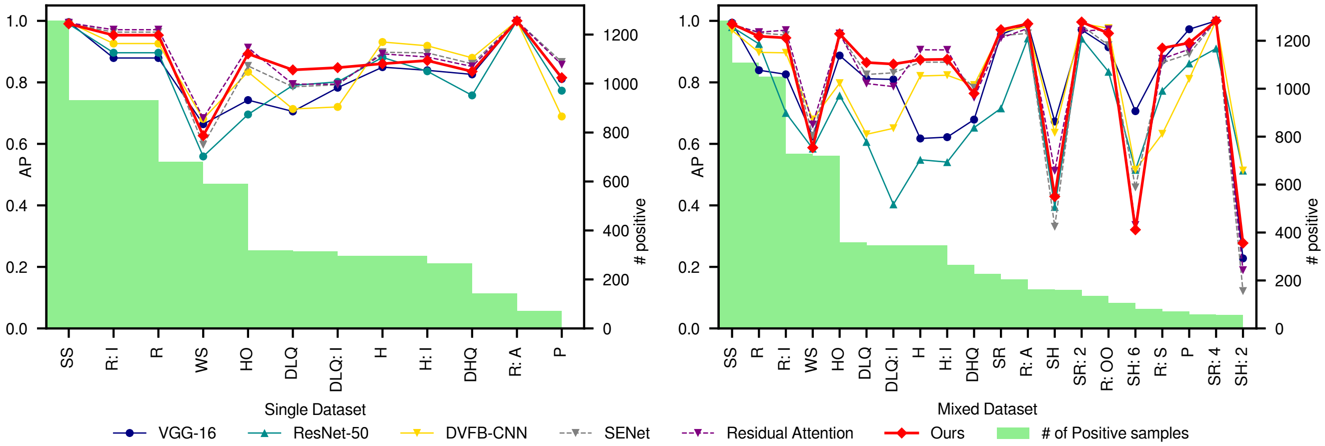


Figure 6. Comparison of transfer learning performance using different deep learning methods, on experimental dataset.

4.4. Transfer Learning with Experimental Dataset

We used the experimental dataset to evaluate transfer learning. The reason to perform transfer learning is that real experimental data is not enough for training a CNN. This is a common hurdle for many specific applications and datasets. For example, our experimental dataset has less than 3,000 images. Therefore, to train on a bigger set of synthetic data is a crucial strategy to actually use the CNN.

We input the images into the trained network from Section 4.2 and computed the global average pooling layer as feature vectors. Then we normalized them and used them to predict the images' attribute with RBF kernel SVMs. SVM parameters C and γ were determined via cross validation.

We compared our method with some state-of-the-art deep learning methods: VGG-16 [28], ResNet-50 [6] and DVFB-CNN [5], as well as the other attention methods trained in Section 4.3: SENet and residual attention. We followed the same experiment setup as in [5] and compared with the APs reported therein, shown in Figure 6. We also list the mAPs in Table 2.

We can conclude from the results that our proposed feature enhancement improves the features further than much deeper networks like ResNet-50; it is even better than DVFB-CNN without precomputed feature transforms or assumption of structural symmetry, and thus our method is more general. AAM shows comparable results among the attention methods which consistently improves the annotations. In particular, AAM has the best mAP in the mixed dataset, showing that multiple spatial attention maps are capable of handling discrepancies of different experimental and imaging setups.

4.5. Classification with Fashion-MNIST

To show AAM is applicable to other tasks and network configurations, we ran classification experiments on Fashion-MNIST. We set up 3 CNNs of different depths to learn to classify the 10 types of clothing articles. The layer configurations are shown in Table 3. We report the prediction accuracy in Table 4. Experiments show improve-

2-layer	3-layer	5-layer
conv-64	conv-64	conv-64 conv-128
(AAM) maxpool		
conv-128	conv-128 conv-256	conv-256 conv-256 conv-256
(AAM) maxpool		
Global Avg Pool fc-10 softmax		

Table 3. Structures of CNNs to predict Fashion-MNIST.

	2-layer	3-layer	5-layer
Original	0.9050	0.9166	0.9294
+AAM	**0.9202**	**0.9269**	**0.9393**

Table 4. Prediction accuracy, on Fashion-MNIST.

ments in accuracy in all of the setups, and thus prove that AAM is equally effective to enhance the features for general datasets.

4.6. Attention Visualization

For qualitative assessment of the attentional features, we computed Grad-CAM [26] to visualize the attribute related activities in our network. We show in Figure 7, from left to right, the input image, and Grad-CAM visualizations of the last convolutional layers, in the networks without attention (VGG-16 + Aggr.), with Normalized CBAM and AAM. We can see the activation regions continue to improve with more sophisticated attention setups. Take (b) as an example: Normalized CBAM and AAM both correctly identify the 2 disjoint high intensity areas that implied 2-fold symmetry, and the Grad-CAM activity of AAM is more precise. This shows that our multiple spatial attention maps can indeed adapt to diverse attributes and react to different features accordingly.

5. Discussions

AAM is a two-step, local-to-global feature enhancement strategy, as proven by the experiments. On one hand, the attentional module reweights the convolutional features at pixel level, maintaining the pixel structure. The result is meaningful local regions being highlighted, evidenced by Section 4.6. On the other hand, the aggregation module reorganizes and summarizes the features by histogram binning. The local pixel structure is not preserved; the result is a representation of non-local feature distributions. Aggregation contributes to better training features with multi-scale and early side supervisions [30]. In summary, attention and aggregation work effectively to encode x-ray scattering image features that are *weak* (local) and *scattered* (non-local).

6. Conclusions

In this paper, we detailed a multi-label visual feature learning framework with the Attentional Aggregation Model. We validated with the experiments that these modules served the purpose of enhancing image features from local to global, which is crucial to understand scientific images with weak, scattered and noisy features.

In the near future, we are interested in finding more effective non-local methods to represent small and "narrow" feature spaces. We are planning to study various graph, correlation and transformation based methods for fine-scale scientific image analysis.

Acknowledgements. This work was supported by NSF IIS-1715985 and NSF IIS-1812606. The authors would like to thank Stony Brook Research Computing and Cyberinfrastructure, and the Institute for Advanced Computational Science at Stony Brook University for access to the high-performance SeaWulf computing system, which was made possible by a $1.4M National Science Foundation grant (#1531492). This research used resources of the Center for Functional Nanomaterials (CFN), and the National Synchrotron Light Source (NSLS) I and II. CFN, NSLS-I and NSLS-II are U.S. Department of Energy (DOE) Office of Science Facilities operated for the DOE Office of Science by Brookhaven National Laboratory under Contract No. DE-SC0012704.

References

[1] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. NetVLAD: CNN architecture for weakly supervised place recognition. In *CVPR*, pages 5297–5307, 2016.

[2] R. Arandjelovic and A. Zisserman. All about VLAD. In *CVPR*, pages 1578–1585, 2013.

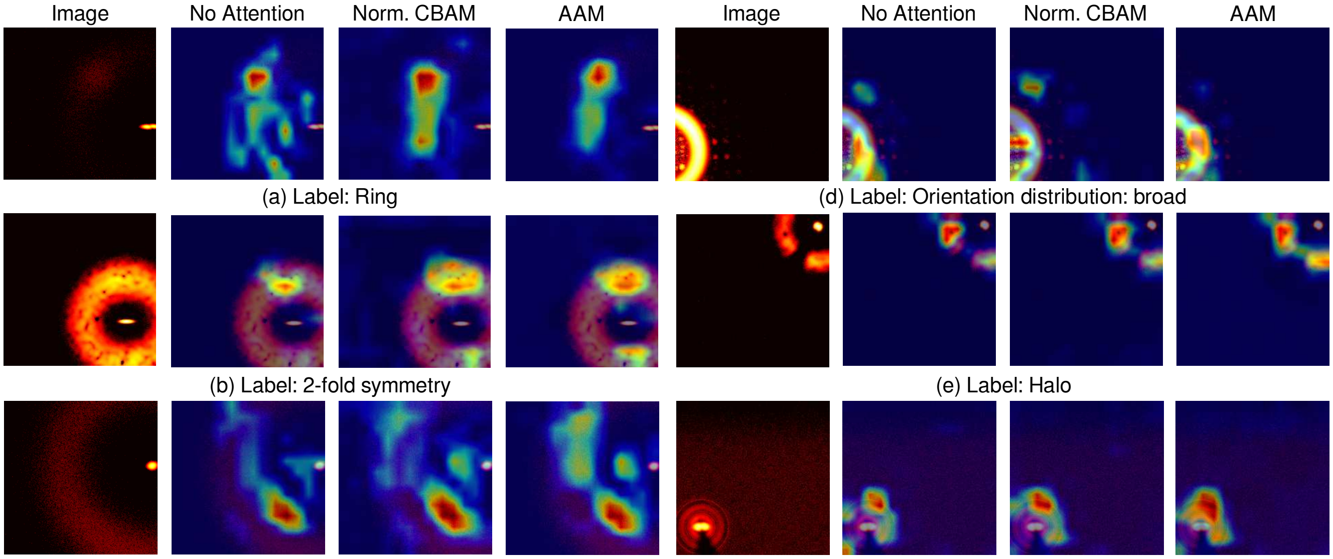
[3] L. Chen, H. Zhang, J. Xiao, L. Nie, J. Shao, W. Liu, and T.-S. Chua. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In *CVPR*, pages 5659–5667, 2017.

[4] Y. Gong, L. Wang, R. Guo, and S. Lazebnik. Multi-scale orderless pooling of deep convolutional activation features. In *ECCV*, pages 392–407, 2014.

[5] Z. Guan, H. Qin, K. Yager, Y. Choo, and D. Yu. Automatic x-ray scattering image annotation via double-view fourier-bessel convolutional networks. In *BMVC*, 2018.

[6] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[7] S. Honari, J. Yosinski, P. Vincent, and C. Pal. Recombinator networks: Learning coarse-to-fine feature aggregation. In *CVPR*, pages 5743–5752, 2016.



(a) Label: Ring

(d) Label: Orientation distribution: broad

(b) Label: 2-fold symmetry

(e) Label: Halo

(c) Label: Width: broad

(f) Label: Isotropic

Figure 7. Grad-CAM visualization of x-ray scattering image samples.

[8] T. Hou, X. Hou, M. Zhong, and H. Qin. Bag-of-feature-graphs: a new paradigm for non-rigid shape retrieval. In *ICPR*, pages 1513–1516, 2012.

[9] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In *CVPR*, pages 7132–7141, 2018.

[10] H. Huang, S. Yoo, K. Kaznatcheev, K. G. Yager, F. Lu, D. Yu, O. Gang, A. Fluerasu, and H. Qin. Diffusion-based clustering analysis of coherent x-ray scattering patterns of self-assembled nanoparticles. In *ACM SAC*, pages 85–90, 2014.

[11] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[12] H. Jégou, M. Douze, C. Schmid, and P. Pérez. Aggregating local descriptors into a compact image representation. In *CVPR*, pages 3304–3311, 2010.

[13] M. H. Kiapour, K. Yager, A. C. Berg, and T. L. Berg. Materials discovery: Fine-grained classification of x-ray scattering images. In *WACV*, pages 933–940, 2014.

[14] H. Larochelle and G. E. Hinton. Learning to combine foveal glimpses with a third-order boltzmann machine. In *NeurIPS*, pages 1243–1251, 2010.

[15] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, volume 2, pages 2169–2178, 2006.

[16] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[17] T.-Y. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie. Feature pyramid networks for object detection. *CVPR*, pages 936–944, 2017.

[18] D. Linsley, D. Scheibler, S. Eberhardt, and T. Serre. Global-and-local attention networks for visual recognition. *arXiv:1805.08819*, 2018.

[19] A. Miech, I. Laptev, and J. Sivic. Learnable pooling with context gating for video classification. *arXiv:1706.06905*, 2017.

[20] A. Newell, K. Yang, and J. Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, pages 483–499, 2016.

[21] W. B. Park, J. Chung, J. Jung, K. Sohn, S. P. Singh, M. Pyo, N. Shin, and K.-S. Sohn. Classification of crystal structure using a convolutional neural network. *IUCrJ*, 4(4):486–494, 2017.

[22] J. S. Pedersen. Analysis of small-angle scattering data from colloids and polymer solutions: modeling and least-squares fitting. *Adv. Colloid Interface Sci.*, 70:171–210, 1997.

[23] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *CVPR*, pages 1–8, 2007.

[24] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *ECCV*, pages 143–156, 2010.

[25] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pages 234–241, 2015.

[26] 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 *ICCV*, pages 618–626, 2017.

[27] A. J. Senesi and B. Lee. Small-angle scattering of particle assemblies. *J. Appl. Crystallogr.*, 48(4):1172–1182, 2015.

[28] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*, 2014.

[29] J. Sivic and A. Zisserman. Video google: a text retrieval approach to object matching in videos. In *ICCV*, pages 1470–1477 vol.2, 2003.

[30] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, pages 1–9, 2015.

[31] Y. Tang, N. Srivastava, and R. R. Salakhutdinov. Learning generative models with visual attention. In *NeurIPS*, pages 1808–1816, 2014.

[32] B. Wang, K. Yager, D. Yu, and M. Hoai. X-ray scattering image classification using deep learning. In *WACV*, pages 697–704, 2017.

[33] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang. Residual attention network for image classification. In *CVPR*, pages 3156–3164, 2017.

[34] Q. Wang, O. Ronneberger, and H. Burkhardt. Fourier analysis in polar and spherical coordinates. *Albert-Ludwigs-Universität Freiburg, Institut für Informatik*, 2008.

[35] S. Woo, J. Park, J.-Y. Lee, and I. So Kweon. Cbam: Convolutional block attention module. In *ECCV*, pages 3–19, 2018.

[36] H. Xiao, K. Rasul, and R. Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv:1708.07747*, 2017.

[37] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, pages 2048–2057, 2015.

[38] K. G. Yager, J. Lhermitte, D. Yu, B. Wang, Z. Guan, and J. Liu. Dataset of synthetic x-ray scattering images for classification using deep learning. 2017.

[39] K. G. Yager, Y. Zhang, F. Lu, and O. Gang. Periodic lattices of arbitrary nano-objects: modeling and applications for self-assembled systems. *J. Appl. Crystallogr.*, 47(1):118–129, 2014.

[40] M. L.-J. Yann and Y. Tang. Learning deep convolutional neural networks for x-ray protein crystallization image analysis. In *AAAI*, pages 1373–1379, 2016.

[41] C. H. Yoon, P. Schwander, C. Abergel, I. Andersson, J. Andreasson, A. Aquila, et al. Unsupervised classification of single-particle x-ray diffraction snapshots by spectral clustering. *Opt. Express*, 19(17):16542–16549, 2011.

[42] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena. Self-attention generative adversarial networks. *arXiv:1805.08318*, 2018.

[43] W. Zhu, Y. Huang, L. Zeng, X. Chen, Y. Liu, Z. Qian, N. Du, W. Fan, and X. Xie. AnatomyNet: Deep learning for fast and fully automated whole-volume segmentation of head and neck anatomy. *Medical physics*, 46(2):576–589, 2019.