Deep Collocative Learning for Immunofixation Electrophoresis Image Analysis

Xiao-Yong Wei, *Senior Member, IEEE*, Zhen-Qun Yang, Xu-Lu Zhang, Ga Liao, Ai-Lin Sheng, S. Kevin Zhou, *Fellow, IEEE*, Yongkang Wu, and Liang Du

***Abstract*— Immunofixation Electrophoresis (IFE) analysis is of great importance to the diagnosis of Multiple Myeloma, which is among the top-9 cancer killers in the United States, but has rarely been studied in the context of deep learning. Two possible reasons are: 1) the recognition of IFE patterns is dependent on the co-location of bands that forms a binary relation, different from the unary relation (visual features to label) that deep learning is good at modeling; 2) deep classification models may perform with high accuracy for IFE recognition but is not able to provide firm evidence (where the co-location patterns are) for its predictions, rendering difficulty for technicians to validate the results. We propose to address these issues with collocative learning, in which a collocative tensor has been constructed to transform the binary relations into unary relations that are compatible with conventional deep networks, and a location-label-free method that utilizes the Grad-CAM saliency map for evidence backtracking has been proposed for accurate localization. In addition, we have proposed Coached Attention Gates that can regulate the inference of the learning to be more consistent with human logic and thus support the evidence backtracking. The experimental results show that the proposed method has obtained a performance gain over its base model ResNet18 by 741.30% in IoU and also outperformed popular deep networks of DenseNet, CBAM, and Inception-v3.**

***Index Terms*— Immunofixation Electrophoresis; deep collocative learning; coached attention gates.**

Manuscript received February 3, 2021; revised March 13, 2021; accepted March 21, 2021. Date of publication March 24, 2021; date of current version June 30, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 61872256 and Grant 81772275 and in part by the Science and Technology Department of Sichuan Province under Grant 2020YFS0125. *(Xiao-Yong Wei and Zhen-Qun Yang are co-first authors.) (Corresponding authors: Yongkang Wu; Liang Du.)*

Xiao-Yong Wei, Xu-Lu Zhang, and Ai-Lin Sheng are with the College of Computer Science, Sichuan University, Chengdu 610065, China, and also with the Center for Artificial Intelligence, Peng Cheng Laboratory, Shenzhen 518055, China (e-mail: cswei@scu.edu.cn).

Zhen-Qun Yang is with the Department of Biomedical Engineering, The Chinese University of Hong Kong, Hong Kong (e-mail: jessicayang@cuhk.edu.hk).

Ga Liao is with the State Key Laboratory of Oral Diseases, National Clinical Research Center for Oral Diseases, West China Hospital of Stomatology, Sichuan University, Chengdu 610041, China.

S. Kevin Zhou is with the Medical Imaging, Robotics, and Analytic Computing Laboratory and Engineering (MIRACLE), School of Biomedical Engineering, University of Science and Technology of China, Suzhou 215123, China, also with the Suzhou Institute for Advance Research, University of Science and Technology of China, Suzhou 215123, China, and also with the Key Laboratory of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing 100190, China.

Yongkang Wu is with the Department of Laboratory Medicine and Outpatient, West China Hospital, Sichuan University, Chengdu 610041, China (e-mail: vipwyk@163.com).

Liang Du is with the Medical Device Regulatory Research and Evaluation Center, West China Hospital, Sichuan University, Chengdu 610041, China, and also with the Chinese Evidence-Based Medicine Center, West China Hospital, Sichuan University, Chengdu 610041, China (e-mail: duliang@scu.edu.cn).

Digital Object Identifier 10.1109/TMI.2021.3068404

I. Introduction

MULTIPLE Myeloma is a malignant disease with 5-year survival rate as low as 53.2% [1] and it is among the top-9 cancer killers (and top-2 among all neoplastic diseases of the blood) in the United States [2], [3]. Immunofixation Electrophoresis (IFE), as one of the mandatory tests for multiple myeloma suspects, is of great importance to clinic diagnosis. IFE is a procedure to identify monoclonal proteins (M-proteins, makers for multiple myeloma) from human serum (or urine), where as shown in Fig. 1, the serum proteins are separated into lanes by electrophoresis and treated with specific antiserum against IgG, IgA, IgM, kappa, and lambda. A precipitin (dark or dense) band will form if a M-protein presents. By observing the co-location (horizontal alignment) of bands between heavy chain lanes (G, A, and M) and light chain lanes (κ and λ),[1] the M-protein can be identified (see examples in Fig. 1).

While the general rule of IFE interpretation by observing co-location is easy to grasp for beginners, the learning curve of IFE interpretation, however, is a typical diminishing-returns curve where the rate of progression increases rapidly at the beginning and decreases over time. Expertise can only be built through a significant amount of case studies because the appearance of bands (even for the same protein present) can vary significantly depending on test conditions (see examples in Fig. 1). Nevertheless, results can be subject to misinterpretation if analysed by junior technicians.

Image recognition with machine learning has been considered as a solution to handle the variety of large-scale samples and provide objective and quality results. In contrast to its successful applications to a wide range of medicinal images

[1] SP lane is a mixed result of multiple proteins and thus with more complex pattern than other lanes. A technician mainly refer to the other 5 lanes and uses SP as a reference. Therefore, for simplicity, the discussion in this study is given on the other 5 lanes without mentioning SP (unless necessary).

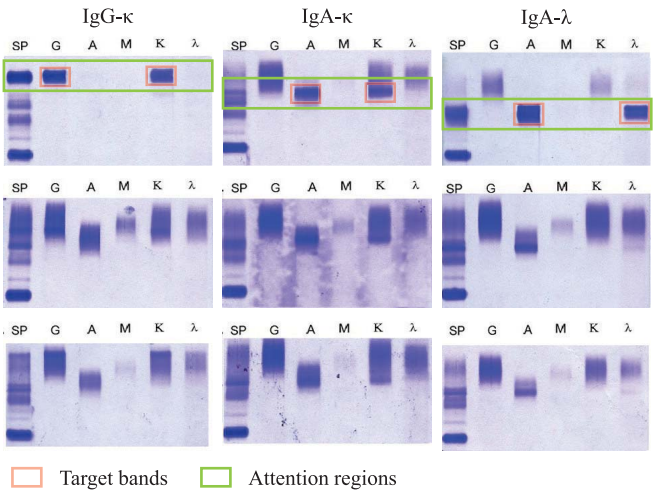

Fig. 1. IFE examples. The first row: samples are with easy-to-recognize co-location patterns; the second and third rows: samples are with the same proteins but are of large variety. Target bands are bands that from the co-location patterns with which the technician identify the types of the proteins. An attention region is defined as the consecutive rows which includes the co-location of target bands, with which the technician verifies if the target bands are prominent enough compared to the other empty or non-target bands within the region.

such as MRI [4], CT [5], and ultrasound images [6], its application to IFE has rarely been found in literature. To construct an IFE solution, we can model it as a classification problem in which we can use the protein presence as class labels and popular models like VGG [7] or ResNet [8] as learners. However, as these models do not learn the locations of patterns explicitly, the models cannot tell which patterns the results are referring to. This is critical in clinic diagnosis scenario where a technician has to verify the exact co-location of bands that the final label is referring to, or a trainee needs to know the mapping between patterns and results from which she can gain knowledge and insights. A commonly adopted way to model the location explicitly is to consider it as an object detection problem to which models like Faster-RCNN [9], Mask-RCNN [10], Yolo [11], or UNet [12] can be applied. Unfortunately, these models require the location labels which are missing from or costly to obtain in history IFE datasets. Nonetheless, an IFE "location" reference is indeed a binary relation (i.e., co-location of bands) instead of a unary relation (i.e., the location of an object) that is usually modeled in conventional models.

In this paper, we address the aforementioned issues by proposing a location-label-free framework for IFE recognition which outperforms the state-of-the-art models while being able to provide location references for the results. To the best of our knowledge, this is the first study of computer-assisted automatic IFE analysis. The rationale behind and advantages of the proposed are as follows.

- *Relation Transformation:* As show in Fig. 2, instead of using the entire image as the input, we feed the deep networks a collocative tensor that encapsulates the co-locations of strips (subdivisions of IFE lanes, see Fig. 3 for details). This transforms the binary relation into unary relation, which makes the input compatible to conventional networks in the way that the elements of the tensor are analog to the image pixels.
- *Guided Inference:* We propose Coached Attention Gates (CAGs) to guide the learning to concentrate more on the direct evidence (i.e., consistency of target bands) rather than the indirect evidence (i.e., inconsistency between target bands and non-target-bands (and empty spaces)). CAG is a formulated way of prior knowledge injection which makes the inference more consistent with the logic of human technicians.
- *Evidence Backtracking:* We propose a Grad-CAM-based evidence backtracking method which back-propagates the CAGs-filtered gradients to the collocative tensor to search for the evidence of the results, and thus localizes the co-locations of the target bands for intuitive result verification. Meanwhile, it eliminates the requirements for location labels in conventional object detection tasks.

II. Related Work

Even though medical image analysis has been a popular topic for decades, the recent advances of deep learning further bring it to a new stage of prosperousness [13]. Most of the existing methods, therefore, are based on deep convolutional neural networks (CNNs) [7]–[12], [14]. Among numerous methods having been proposed, we will focus on the classification models [15]–[20] and detection/segmentation[2] models [21]–[30] that are related to this paper. The reader can refer to [13], [31] for a more comprehensive survey.

A. Classification Models

Classification models are built for identifying the types of samples or lesions so as to provide references for disease diagnosis. A commonly adopted way is to conduct transfer learning that starts with a well-recognized CNN model developed in computer vision (e.g., VGG [7], ResNet [8], Inception-v3 [14]) and trained on public datasets such as ImageNet [32], CoCo [33], and CIFAR-10 [34]. The model will then be fine-tuned on the target medical image dataset. Works following this paradigm include these using VGG for retina (FUNDUS) [15] or spine (MRI) [16] images, RestNet for skin [17] or heart [18] images, and Inception-v3 for retina [19] and skin [20] images.

While these classification models are related to IFE recognition in the way that both are modeling the mapping from a medical image to labels, the evidence for inference, however, is different. In classification models, the evidence usually includes visual features (e.g., colors, textures, shapes) of a target. By contrast, in IFE recognition, visual features (the darkness or density) of a target (band) are not the first priority for inference unless it horizontally collocates with another band of similar visual features. For the same reason, the way for a human technician to identify an IFE type is to locate

[2]Detection and segmentation models are two different kinds of models in literature. We put them together for easier description because in this paper we focus on object localization which they share in common. Their differences are beyond the topic of this paper.

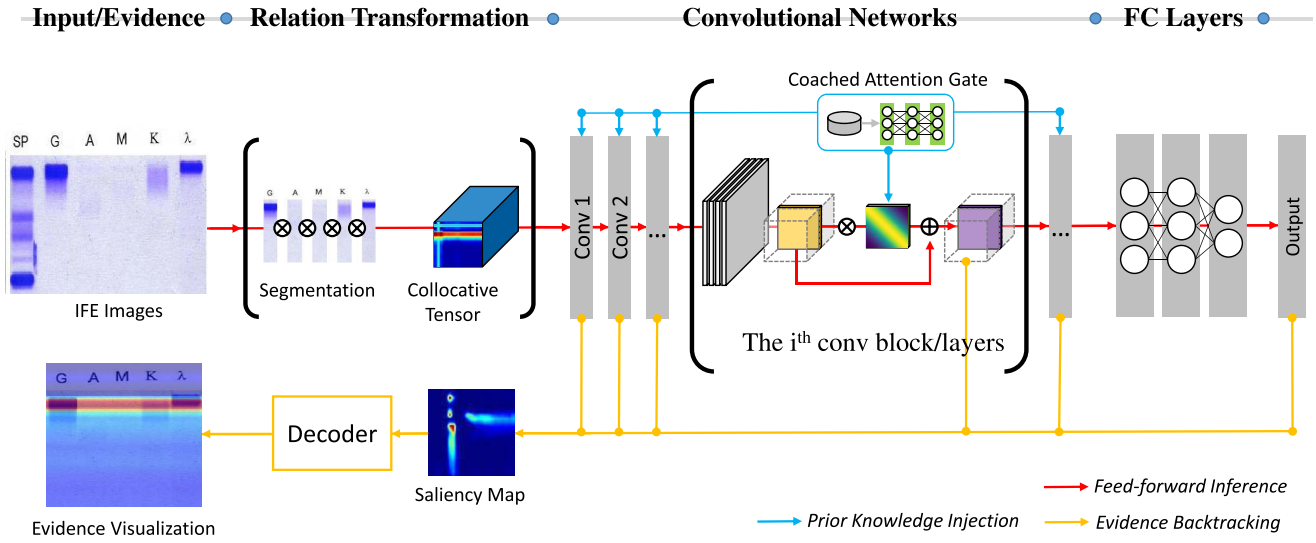

Fig. 2. Collocative learning framework.

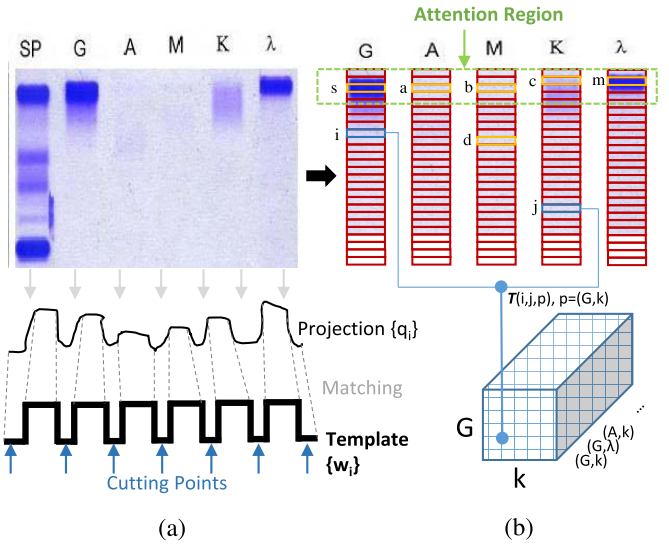

Fig. 3. Relation transformation: (a) Lane segmentation, (b) Strip division and collocative tensor construction.

a band in one lane first and then search over the other lanes to see if there is another similar band appearing at the similar horizontal position. In other words, the pattern forming in IFE is based on the inter-target relation, which is not well modeled by these CNN-based classification models. We will use the collocative tensor to address this issue. More details will be given in Section III.

B. Detection/Segmentation Models

In IFE recognition, if we need to know which co-location pattern has led to the result, we may have to use detection/segmentation models. These models are built for identifying objects and their locations in the images. Intensive studies have been conducted for general images, resulting in promising models like Faster-RCNN [9], Mask-RCNN [10], Yolo [11], and UNet [12]. These models have also been widely adopted such as Faster-RCNN for intervertebral disc detection [21] and lung nodules detection [22], Mask-RCNN in lung segmentation [23] and follicle segmentation [24], Yolo for detecting of breast masses [25] and lung nodules [26], and UNet for detections in kidney, chest, liver, abdomen, brain images [12], [27]–[30].

While promising results have been obtained, it is not feasible to apply these models to IFE because they model a single-object-location relation, which is different from the co-location (of objects) relation in IFE. Nonetheless, they all require location labels (e.g., rectangles or masks of the target objects/regions) which are not easy to obtain in IFE images. To bypass the requirement for location labels, we have adopted Grad-CAM [35] and attention techniques [36]–[38] for IFE in this paper.

C. Grad-CAM and Attentions

Grad-CAM [35] and its alternatives Grad-CAM++ [39] and CAM [40] are techniques to generate "visual explanation" for the resulting label of a CNN-based model. It back-propagates, summarizes, and visualizes the gradients to a certain layer of the network for easier observation that which part of the layer has been activated during feed-forward inference. This often leads to a heatmap aligned with the target region (e.g., a dog or a cat) which can therefore also be treated as a rough location of the target. However, the broadly spreading heatmap is far beyond an accurate location required by IFE analysis. We will use the collocative tensor and attention mechanism to create constraints for forward propagation so as to guide the inference to be conducted in a more concentrated way and thus generate a more accurately distributed heatmap for localization.

In this context, this paper is also related to attention mechanisms [36]–[38] which are usually employed to encourage the CNN models to focus on certain channels or spatial regions. Representative works include SENet [36], CBAM [37], and

Non-local Net [38]. These attention mechanisms are adopted mainly for improving the performance of the models in terms of classification/detection accuracy. By contrast, the coached attention gate proposed in this paper is an attention method for inference regulation rather than accuracy enhancement.

III. Deep Collocative Learning for IFE

In this section, we describe our methodology for constructing an end-to-end deep learning framework for IFE analysis. The framework is called Deep collocative learning which answers three questions: *1) How do we construct a collocative tensor to make the input compatible with deep classification models? 2) How do we guide the inference to make it consistent with the human logic? and 3) How do we backtrack the co-location pattern that leads to the final result?*

A. Collocative Tensor for Relation Transformation

The strength of deep classification models is to learn unary relations that map a pattern (a group of visual features) to a target label. In IFE analysis, however, human technicians have to learn binary relations that map the co-location of two patterns (visual features of two bands) to a protein label. We propose to construct a collocative tensor for transforming the binary relations into unary ones.

1) Slicing IFE Into Lanes and Strips: To construct the tensor, we have to slice an IFE into lanes (Fig. 3a) and then cut each lane into strips (Fig. 3b). It is easy to separate lanes if they are at fixed positions and with fixed widths. Data acquired from real applications with different devices, however, is not the case. We can transform it into a dynamic programming problem to deal with this issue. Let us build a template series $W = \{w_i\}$ with its length equal to the average width of all IFE, and with 6 plateaus and 7 valleys representing the positions of lanes and spaces, and project an IFE vertically to form another series $Q = \{q_i\}$. We can then find the positions of lanes as long as we find the best match between these two series of W and Q. This is a typical dynamic programming problem which can be solved by Dynamic Time Warping algorithm [41], [42]. After the best match has been found, we cut the IFE at the centers of valleys to separate lanes and slice each lane into n strips with equal height. In a similar way, we can separate the lanes from the heading characters.

2) Tensor Construction: With the strips, a collocative tensor $\mathcal{T}$ can be composed with inter-strip correlations. As an IFE co-location is defined as the horizontal alignment of two bands from a heavy chain and a light chain respectively, let us organize lanes into pairs of heavy chains $\{G, A, M\}$ and light chains $\{\kappa, \lambda\}$ which results in a set

$$\mathcal{P} = \{(G,\kappa),(G,\lambda),(M,\kappa),(M,\lambda),(A,\kappa),(A,\lambda),(G,A),(G,M),(A,M),(\kappa,\lambda)\} \quad (1)$$

consisting of all the possible pairs from which a technician searches for the collocation of bands. The $(i, j, p)^{th}$ element of the tensor, which models the correlation of the i^{th} and the j^{th} strips from the p^{th} lane pair, can thus be written as

$$\mathcal{T}(i, j, p) = \Phi^p(s_i, s_j) \cdot \Psi^p(s_i, s_j) \cdot \pi^p(s_i, s_j) \quad (2)$$

which is a joint consideration of the similarity $\Phi^p(s_i, s_j)$ between, average density (darkness) $\Psi^p(s_i, s_j)$ of, and co-location $\pi^p(s_i, s_j)$ of two strips. The $\Phi^p(s_i, s_j)$ can be easily implemented by adopting from popular metrics of visual features such as the *cosine similarity* or *Euclidian distance*, while so as for the $\Psi^p(s_i, s_j)$ which can be simply the average gray level of the bands. The $\pi^p(s_i, s_j)$ is an indication of how close two strips co-located vertically as

$$\pi^k(s_i, s_j) = \frac{1}{\sqrt{2\pi}\cdot 1}\exp\left(-\frac{1}{2}\left(\frac{\frac{(3\cdot 1)|i-j|}{n-1}}{1}\right)^2\right) = \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{9}{2}\left(\frac{|i-j|}{n-1}\right)^2\right), \quad (3)$$

which is indeed a Gaussian normalized vertical distance of the two target strips. The normalization process is equal to applying the weights in the range of [0, 3] times of its standard deviation of a Gaussian ($X \sim N(0, 1)$) as a mask on the strip distance $\frac{|i-j|}{n-1}$. The number 3 (i.e., three times of the standard deviation) is both a commonly adopted setting and an optimal number in our empirical study. The normalization process is also a regulator to have the resulting tensor in favor of strips that are vertically close to each other, because they are high potentials to form co-location bands.

After the relation transformation, the collocative tensor $\mathcal{T}$ is then compatible with any of the CNN-based classification models, in the way that $\mathcal{T}$ is a tensor of dimensions $n \times n \times \|\mathcal{P}\|$ where the first two dimensions ($n \times n$) are imitating the spatial layout of an image and the last one $\|\mathcal{P}\|$ imitating the channels. By using the protein labels (given by technicians) in the history data, we can easily conduct a classification learning with sophisticated models like VGG, ResNet, or Inception.

B. Guided Inference With Coached Attention Gates

1) Deep Classification Vs. Location Inference: Deep classification models are often employed as black boxes, because their implicit way for inference is not necessarily consistent with the human logic. Taking Fig. 3 as an example, to identify a co-location pattern in IFE, a human technician is looking for evidence mainly from the attention region (e.g., the dashed green box in Fig. 3b) where the strips from two different lanes that are with similar densities and distributed vertically close to each other (e.g., the s and m in the target lanes G and κ respectively). A technician may also compare these strips with the adjacent ones (e.g., the a, b and c) to verify if they (the s and m) are salient enough to form a target band and also recognize the height of the band.

While the logic is intuitive, it is not the only option. For example, instead of verifying a co-location directly by comparing strips in the attention region, a co-location can be identified indirectly if the strips in the non-attention region (e.g., d) are not forming a co-location pattern (with significant dissimilarities) with strips in the attention region (e.g., s and m). In deep networks, this type of evidence will be brought into the model, because the dissimilarities between the strips from the attention and the non-attention regions will form prominent

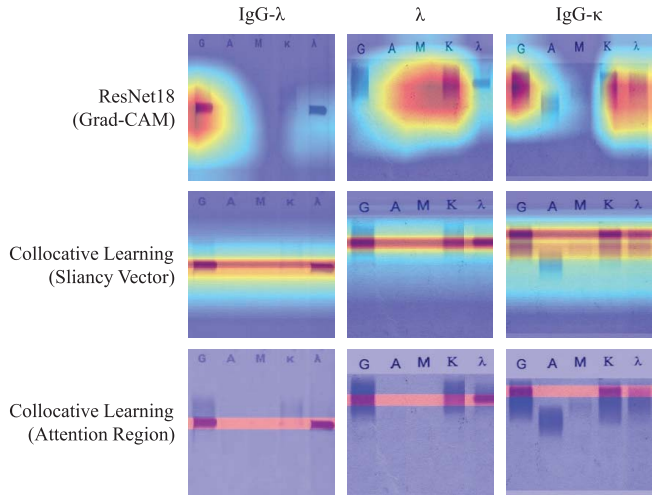

Fig. 4. Saliency maps for evidence visualization generated by mixed inference in conventional CNNs (the first row) and guided inference in collocative learning (the second and third rows) respectively, where the samples in the second row are generated using the saliency vector (Eqn. (11)) and samples in the third row are generated by exact attention region localization (Eqn. (12)).

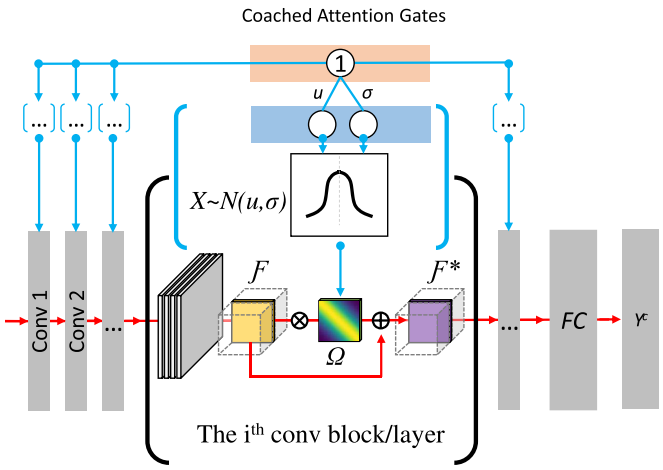

Fig. 5. Coached Attention Gates (CAGs) is a sub-network (blue lines in the figure), where an attention gate Ω will be generated and integrated over the last feature map $\mathcal{F}$ of each convolutional block/layer. The μ and σ associated with each gate generator are the mean and standard deviation of a Gaussian distribution $X \sim N(\mu, \sigma)$ which will be used to generate the mask Ω that puts its attention on the diagonal neighborhood.

"edges" in the collocative tensor which can easily be identified through convolution. Similarly, the formation of a band can also be observed in an indirect way through the comparison with non-adjacent strips in the non-attention regions. The pooling process can bring such information into the model.

While the mixed inference with both direct and indirect evidence in conventional CNNs has long been considered as an advantage for a more flexible and comprehensive reasoning, it makes the resulting models less intuitive. An illustration to demonstrate this issue can be found in Fig. 4, where we compare the Grad-CAM visualization results (saliency maps) of models trained with conventional CNNs and the guided inference respectively. We can see that in conventional deep networks, the evidence used for inference are not necessarily focusing on the attention regions (on which human technician put their focus).

2) Coached Attention Gates: To make the inference more consistent with the human logic, we propose to "guide" the inference with *Coached Attention Gates (CAGs)*. The motivation is to force the deep models put more attention on the IFE regions that are with high potential to be an attention region. This can be easily implemented with the collocative tensor, because the strips from potential attention regions are vertically adjacent to each other and should thus locate in the diagonal neighborhood of the "spatial" dimensions of the tensor. Therefore, we can set attention masks, which focus on the diagonal neighborhood, on the spatial dimensions of the feature maps generated during the feed-forward propagation to force the inference to concentrate on the direct evidence more than the indirect ones.

Beyond the intuition that these masks can be simply generated by a Gaussian along the diagonal, there are a set of parameters needed to be tuned before it becomes a practical mechanism. It includes parameters that define the shape of the Gaussian, to which layers the attention masks to be applied, and the degree that the masks affect the feature maps. Instead of searching optimal parameters through tedious empirical studies, we can leverage the power of neural networks to figure them out automatically.

As shown in Fig. 5, the *Coached Attention Gates* (CAGs) is a network consisting of 3 layers with: 1) the first layer composed with a single node of a fixed value 1, 2) the second layer composed with $2 \times L$ (L is the number of blocks/layers that need attention guidance) nodes which are fully connected to the nodes in the first layer with weights μ and σ, and 3) the third layer composed with L attention masks (matrices) in which each of them is with $m \times m$ nodes. We organize every 2 nodes in the second layer and a mask from the third layer as a gate generator in which the nodes from second and third layers are fully connected. There is no connection across generators. This results in L gate generators with each of them assigned to a block/layer for generating an attention gate mask $\mathbf{\Omega}$ over the feature map $\mathcal{F}$ (with shape $m \times m \times Z$) of the block/layer.

The rationale behind the CAGs is that the μ and σ associated with each gate generator is representing the mean and standard deviation of a Gaussian distribution $X \sim N(\mu, \sigma)$ which we will use to generate the mask $\mathbf{\Omega}$ that puts its attention on the diagonal neighborhood. By applying this mask to the spatial dimensions of the feature map $\mathcal{F}$, we can guide the feed-forward inference focuses more on the attention regions of the IFE. This can be denoted as

$$\mathcal{F}_k^* = \mathcal{F}_k \bigotimes \mathbf{\Omega} \bigoplus \mathcal{F}_k \tag{4}$$

where $\bigotimes$ and $\bigoplus$ are element-wise multiplication and addition respectively, and $0 \leq k \leq Z$ is an iterator for traversing channels of $\mathcal{F}$ and the resulting feature map $\mathcal{F}^*$. During the back-propagation, the μ and σ will updated automatically so that the shape of the Gaussian and $\mathbf{\Omega}$ will be updated. This generates different masks at different blocks/layers and also controls the degree of attention at each block/layer adaptively.

To this end, instead of using weighted sum and activation functions, the feed-forward function between the second and third layers is defined as

$$\begin{aligned}\Omega_{ij} &= \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(X_{ij}-\mu)^2}{2\sigma^2}\right),\\ X_{ij} &= \mu_0 + \frac{3\sigma_0|i-j|}{m-1},\end{aligned} \tag{5}$$

where μ_0 and σ_0 are initial values for μ and σ respectively, which defines a Gaussian distribution $X \sim N(\mu_0, \sigma_0)$ for initializing the mask. Meanwhile, we have to modify the back-propagation functions from the third to the second layer as well. According to the chain rule, the gradients at the $(i,j)^{th}$ node of Ω regarding current prediction Y^c (for class c) can be defined as

$$\begin{aligned}\frac{\partial Y^c}{\partial \mu} &= \sum_{i,j}\frac{\partial Y^c}{\partial \Omega_{ij}}\frac{\partial \Omega_{ij}}{\partial \mu}\\ &= \sum_{i,j}\frac{\partial Y^c}{\partial \Omega_{ij}}\frac{\partial\left[\frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(X_{ij}-\mu)^2}{2\sigma^2}\right)\right]}{\partial \mu}\\ &= \sum_{i,j}\frac{\partial Y^c}{\partial \Omega_{ij}}\left[\frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(X_{ij}-\mu)^2}{2\sigma^2}\right)\left(-\frac{2(X_{ij}-\mu)}{2\sigma^2}\right)\right]\\ &= \frac{1}{\sqrt{2\pi}\sigma^3}\sum_{i,j}\frac{\partial Y^c}{\partial \Omega_{ij}}(X_{ij}-\mu)\exp\left(-\frac{(X_{ij}-\mu)^2}{2\sigma^2}\right)\\ &= \frac{1}{\sigma^2}\sum_{i,j}\frac{\partial Y^c}{\partial \Omega_{ij}}(X_{ij}-\mu)\Omega_{ij},\end{aligned} \tag{6}$$

$$\begin{aligned}\frac{\partial Y^c}{\partial \sigma} &= \sum_{i,j}\frac{\partial Y^c}{\partial \Omega_{ij}}\frac{\partial \Omega_{ij}}{\partial \sigma}\\ &= \sum_{i,j}\frac{\partial Y^c}{\partial \Omega_{ij}}\frac{\partial\left[\frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(X_{ij}-\mu)^2}{2\sigma^2}\right)\right]}{\partial \sigma}\\ &= \sum_{i,j}\frac{\partial Y^c}{\partial \Omega_{ij}}\left[-\frac{1}{\sqrt{2\pi}\sigma^2}\exp\left(-\frac{(X_{ij}-\mu)^2}{2\sigma^2}\right)\right.\\ &\quad\left.+\frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(X_{ij}-\mu)^2}{2\sigma^2}\right)\left(-\frac{2(X_{ij}-\mu)^2}{2\sigma^3}\right)\right]\\ &= \frac{1}{\sqrt{2\pi}\sigma^2}\sum_{i,j}\frac{\partial Y^c}{\partial \Omega_{ij}}\left(\frac{(X_{ij}-\mu)^2}{\sigma^2}-1\right)\exp\left(-\frac{(X_{ij}-\mu)^2}{2\sigma^2}\right)\\ &= \frac{1}{\sigma}\sum_{i,j}\frac{\partial Y^c}{\partial \Omega_{ij}}\left(\frac{(X_{ij}-\mu)^2}{\sigma^2}-1\right)\Omega_{ij}\end{aligned} \tag{7}$$

where $\frac{\partial Y^c}{\partial \Omega_{ij}}$ and $\frac{\partial Y^c}{\partial \Omega_{ij}}$ can be learned from the standard back-propagation.

C. Location Evidence Backtracking

To backtrack the co-location evidence, we need to generate a saliency map for the spatial dimensions of the collocative tensor first, and then "decode" this map to the original IFE image for a human understandable visualization of the location evidence.

1) Saliency Map Generation: We adopt the Grad-CAM [35] which is able to generate a saliency map $\mathcal{S}^{\mathcal{F}^*}$ for any given feature map $\mathcal{F}^*$ (with shape $W \times H \times Z$) regarding current prediction Y^c (for class c). The process is indeed a weighted average of gradient maps over channels of $\mathcal{F}^*$ in which the original definition (in [35]) of the weight for a channel k can be elaborated as

$$w_k^c = \frac{1}{WH}\sum_i^W\sum_j^H\frac{\partial Y^c}{\partial \mathcal{F}^*_{k(i,j)}} \tag{8}$$

where the (i,j) are the iterators for spatial locations and $\frac{\partial Y^c}{\partial \mathcal{F}^*_{k(i,j)}}$ is the gradient at the $(i,j)^{th}$ element of channel k. The saliency map can then be calculated in a location-wise way as

$$\mathcal{S}^{\mathcal{F}^*}_{ij} = \sum_k^Z w_k^c \cdot \mathcal{F}^*_{k(i,j)}. \tag{9}$$

In most of the papers that the Grad-CAM has been employed, the saliency map is generated only for the last layer/block of the network. In this paper, we generate the saliency maps for all layers/blocks and summarize them into a single map. Assume we have L layers/blocks, we implement this with an average pooling as

$$\mathcal{S} = \frac{1}{L}\bigoplus_{l=1}^{L}\mathcal{S}^{\mathcal{F}^*}_{\{l\}} \tag{10}$$

where the subscript l is the layer/block number.

Finally, the saliency map for the spatial dimensions of collocative tensor $\mathcal{T}$ can be obtained by resizing $\mathcal{S}$ to the desired size. Let us denote it as $\mathcal{S}^{\mathcal{T}}$ hereafter.

2) Decoding $\mathcal{S}^{\mathcal{T}}$ for Location Evidence: The salient values in $\mathcal{S}^{\mathcal{T}}$ are indeed for the location pairs. More specifically, a value $\mathcal{S}^{\mathcal{T}}_{ij}$ indicates the degree that the cross-reference of the i^{th} and the j^{th} *rows* of the original IFE image has contributed to the prediction Y^c. Note that we have used the term *rows* to refer to IFE stripes at the same vertical location across lanes (i.e., the i^{th} row is a combination of the i^{th} strips from all lanes) instead of using more finely defined strips. This is due to the fact that CNN learning preserve (even roughly) the spatial dimensions while fusing the channels, a nature inherited from colored image processing. In IFE, the channel imitations are the lane pairs with which, as we have discussed in Section III-A.2, it is reasonable to fuse. Furthermore, checking a row instead of the target stripes alone is consistent with human logic of checking a co-location from all adjacent strips in the attention region. Therefore, a saliency value is a fused contribution indicator of the strips over the related rows.

Intuitively, the more frequent a row has been referred to, the more contributions it has to the prediction. We can turn this intuition into a voting scheme in which we give each saliency value to the two rows related and accumulate values for each row as its final score. The location evidence of an IFE image can thus be generated by visualizing these scores. This process can be simply implemented by the multiplying

TABLE I
LABEL DISTRIBUTION OF IFE DATASET. THE LABEL NON-M IS GIVEN WHEN NONE OF THE M-PROTEINS PRESENTS IN THE SAMPLE

Label	Non-M	IgG-κ	IgG-λ	IgA-κ	IgA-λ	IgM-κ	IgM-λ	κ	λ
$\#samples$	2954	435	392	136	198	78	27	37	95

the $\mathcal{S}^{\mathcal{T}}$ by a columna vector of ones as

$$\vec{\mathbf{s}} = \frac{1}{\|\vec{\mathbf{1}}\|}\left(\mathcal{S}^{\mathcal{T}}\vec{\mathbf{1}} + \left[\mathcal{S}^{\mathcal{T}}\right]^{T}\vec{\mathbf{1}}\right) \tag{11}$$

where $[\cdot]^T$ is the matrix transpose operator and $\vec{\mathbf{s}}$ is the final saliency vector which can be assigned to corresponding rows in the IFE for location evidence visualization (see examples in Fig. 4).

3) Using the Saliency Vector for Attention Region Localization: The final saliency vector $\vec{\mathbf{s}}$ is indicative rather than determinative for the localization of attention area. To find the exact location, we treat $\vec{\mathbf{s}}$ as 1-D signal and transform the problem into finding the best sliding window position where the summation of energy of the $\vec{\mathbf{s}}$ segmentation under the window reaches the maximum. This can be implemented as a 1-D convolution process

$$\begin{aligned} i^* &= \underset{i=1}{\overset{|\vec{\mathbf{s}}^*|}{\operatorname{argmax}}}\, \vec{\mathbf{s}}_i^*, \\ \vec{\mathbf{s}}^* &= \vec{\mathbf{s}} \otimes \vec{\mathbf{w}} \end{aligned} \tag{12}$$

where the $\otimes$ is a standard convolution operator, $\vec{\mathbf{w}}$ is a ones vector with its length equaling to the expected height of the attention region, and the i^* is indeed the position of the center of the optimal window with which we can use as the location of the attention region. The results are shown in Fig. 4.

IV. EXPERIMENTS

A. Setup

1) Dataset: To evaluate the performance, an IFE dataset has been constructed consisting of 4352 IFE images with their protein presence as class labels. The dataset construction has been conducted by the West China Hospital of Sichuan University within the period of October 20^{th}, 2015 though May 11^{th}, 2018.[3] There are 9 labels of IgG-κ, IgG-λ, IgA-κ, IgA-λ, IgM-κ, IgM-λ, κ, λ, and Non-M (i.e., none of the M-proteins presents in the sample) used for labeling. Every image is labeled by 3 $\sim$ 5 technicians. A label will be assigned only if it has reached an agreement with more than 3 technicians. All images are normalized to fit a 200 by 200 pixel box while preserving their aspect ratios.

To complete the IFE segmentation as introduced in Section III-A.1, we have adopted the DTW implementation from *https://github.com/pierre-rouanet/dtw*. Besides, all the source code for collocative learning is available at *https://github.com/lookwei/collocative-learning-4-IFE*.

[3]Samples are in fact collected through the Integrated Care Organization (WCO) which consists of 686 hospitals across West China. West China Hospital is leading the WCO and is in charge of the data aggregation. The dataset is the largest one in literature and the source is heterogeneous which represents a wide diversity of patients, samples, and devices.

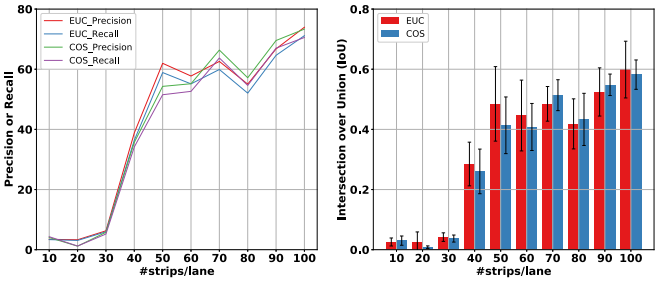

Fig. 6. Performance of the correlation metrics over the n (the number of strips per lane).

2) Evaluation: For evaluation of the performance, we use 10-fold cross validation where in each round, the dataset will be randomly divided into a training and a testing set consisting of 90% and 10% images of the whole dataset, respectively. We have employed the standard F1-score to evaluate the classification of protein presence (labels). To evaluate the performance of evidence backtracking (attention region localization), we use the commonly adopted Intersection over Union (IoU) which measures the ratio of intersection area over the union area of the two regions (i.e., the ground-truth attention region and the detected region). To address the lack of ground-truth for attention regions, we have labeled 10% of the dataset so that the evaluation can be conducted. Nonetheless, since there is no exact localization in convectional classification models, we use the method [39] in which a threshold is used to convert a saliency map (generated with Grad-CAM methods) into an attention region (mask) by keeping only the pixels with saliency values larger than the threshold. For the proposed method, we can detect the exact attention region for every IFE with the approached introduced in Section III-C.3.

In addition, we have also employed the precision and recall of a ground-truth attention region and a detected region to evaluate the localization performance. We will denote them as Precision^R and Recall^R in the experiments.

B. Selection of Base Model

Even the base model selection is not critical in the proposed framework in the way that it is open and compatible with any CNN-based models, we still need a base model to conduct experiments for the investigation over the other parameters. Table II shows the results of a comparison among popular networks such as ResNet18 [8], ResNet50 [8], VGG16 [7], VGG19 [7], Inception-v3 [14]. It is easy to see that ResNet18 has demonstrated a comparable performance over the others (even slightly better than VGG models) while having gained the best efficiency (43.20% $\sim$ 86.98% faster than others). Therefore, we will use ResNet18 as the base model in the following experiments for faster iterations.

C. Collocative Tensor Construction

To construct a collocative tensor, there are two parameters to optimize, namely the number of strips per lane n and the correlation metric to use. For the number of strips per lane n,

TABLE II
PERFORMANCE COMPARISON OF BASE MODELS. THE BEST AND SECOND BEST PERFORMANCE ARE IN BOLD AND ITALIC RESPECTIVELY

Model	F1-score (%)	IoU (%)	Precision^R (%)	Recall^R (%)	Forward (ms)	Backward (ms)
ResNet18 [8]	92.09±1.84	**7.36±2.80**	*34.82±10.06*	**8.84±3.61**	**60.81±3.67**	**167.33±8.53**
ResNet50 [8]	*94.39±1.47*	*5.43±1.46*	26.30±10.65	*6.60±1.73*	131.62±6.71	363.96±15.07
VGG16 [7]	88.73±1.83	3.17±0.78	**44.35±6.94**	3.34±0.83	391.63±17.64	1006.18±30.05
VGG19 [7]	90.82±1.80	2.93±0.79	29.12±6.11	3.14±0.82	467.01±23.01	1182.55±43.28
Inception-v3 [14]	**94.42±3.30**	3.90±2.69	15.05±10.46	5.05±3.31	*107.06±5.23*	*279.43±11.95*

TABLE III
COMPARISON OF PERFORMANCES WITH DIFFERENT COACHING STRATEGIES. THE BEST PERFORMANCE ARE IN BOLD. NOTE WE CALL THE PERFORMANCE OF "$\vartheta^p(.)$ ONLY" AND "$\vartheta^p(.)$ + CAGS" STRATEGIES OF "WITHOUT CAGS" AND "WITH CAGS" IF THE $\vartheta^p(.)$ IS USED BY DEFAULT FOR PREPROCESSING

Strategy	F1-score (%)	IoU (%)	Precision^R (%)	Recall^R (%)
No Coaching	92.87±0.48	53.49±7.11	67.84±6.96	65.75±6.66
$\Psi^p(.)$ Only	93.29±0.78	59.88±9.43	73.96±8.59	71.16±8.21
CAGs Only	**94.78±0.18**	53.96±8.52	67.61±9.54	65.55±9.12
$\Psi^p(.)$ + CAGs	94.20±0.37	**61.92±4.56**	**76.04±4.58**	**72.94±4.53**

we have investigated options from 10 to 100 with a step-size 10, while for the correlation metric, we have studied performance of *cosine similarity* (COS) and *Euclidian distance* (EUC). The results are shown in Fig. 6.

It is easy to see that the $n = 100$ with EUC is an optimal setting with which the collocative learning has obtained a slight $F1$-score gain over the base model ResNet18. Most importantly, the learning has demonstrated a significant superiority by improving the evidence backtracking (indicated through IoU) by 713.59%. We will use $n = 100$ and EUC as a default setting hereafter unless otherwise indicated.

D. Coached Attention Gates (CAGs)

As discussed in Section III-B.2, we have introduced CAGs to guide the inference. In addition, the function $\Psi^p(.)$ in Eqn. (3) is indeed a regulator to force the input (i.e., the collocative tensor) of the networks to assign higher weights for potential co-location patterns. It can be considered as a pre-processing coaching strategy, which works with CAGs jointly for guiding the inference. In this section, we conduct experiments to investigate how the CAGs and $\Psi^p(.)$ work together and separately. The results are shown in Table III.

1) Preprocessing Vs. Online Inference Coaching: Comparing between the preprocessing coaching strategy using $\Psi^p(.)$ and the online strategy using CAGs, when they are used alone, both of them have obtained improvement over the no-coaching baseline in terms of F1-score, a sign that both strategies are of help to the recognition. On the other hand, in terms of IoU, the preprocessing strategy has demonstrated much better performance over CAGs. This is not surprising, because with the regulator $\Psi^p(.)$, the indirect evidence can be filtered out straightforwardly prior to the actual feed-forward inference. By contrast, without the regulator, the CAGs are leaving with a lot of distractions from the indirect evidence when it works alone, and thus it has developed inference skills less intuitive to human logic even it has gained the best F1-score performance among all strategies. However, it is easy to see that, when the two strategies are used at the same time, much better results have been obtained than they are used alone. This is an indication that the preprocessing and online coaching strategies can play complementary roles for the inference.

Since the regulator $\Psi^p(.)$ is straightforward, let us consider it as a default preprocessing for the collocative learning, so we can investigate how CAGs can help the actual inference by comparing between the performances without (i.e., $\Psi^p(.)$ Only) and with the CAGs (i.e., $\Psi^p(.)$+CAGs). For simplicity, we will hereafter call these two performances as "Without CAGs" and "With CAGs".

2) Case Study: The performance gain indicates that the CAGs have successfully guided the learning to focus on the attention regions and avoid distractions. This can be observed more intuitively from the first 4 columns of Fig. 7, where the saliency maps with CAGs are more densely distributed on the attention regions than those without CAGs. Another typical observation is that CAGs, as a formulated constraint to confine the inference process to search for evidence mainly from the diagonal neighbourhood of the collocative tensor, has increased the determinacy of the evidence backtracking. As a result, the multiple attention regions appeared in the saliency maps of some examples without CAGs have been reduced to a single or one dominating attention region (see the 1^{st}, 2^{nd}, and the 4^{th} columns of Fig. 7). This has also been reflected in Table III where standard deviations of the performance have decreased when CAGs have been used.

While promising results have been observed, CAGs do not work for all cases. Taking the 5^{th} and 6^{th} columns of Fig. 7 as an example, we have obtained minor performance gain using CAGs. Most of such samples are "easy" IFE cases in which the target bands are distinct with darker density and well aligned horizontally so that they can be easily identified. By contrast, for samples like the 7^{th} and 8^{th} columns of Fig. 7 in which the target bands are mal-positioned (not well aligned horizontally due to variety of test conditions), the resulting saliency maps with CAGs are more confusing than those without CAGs. However, it is inconclusive that whether and how much we can observe improvement on such samples, given the performance improvement we have obtained on the 2^{nd} (significant) and 5^{th} columns (minor) with similar mal-position.

3) Adaptive Attention: In the experiments, we have set 1 CAG for each of the 8 ResNet18 blocks at the position right after its last 3×3 convolution layer, in the hope that the shape of the CAG will be formed adaptively to the corresponding block. A visualization of the resulting CAGs is given in Fig. 8. We can see the peak weights of the CAG masks

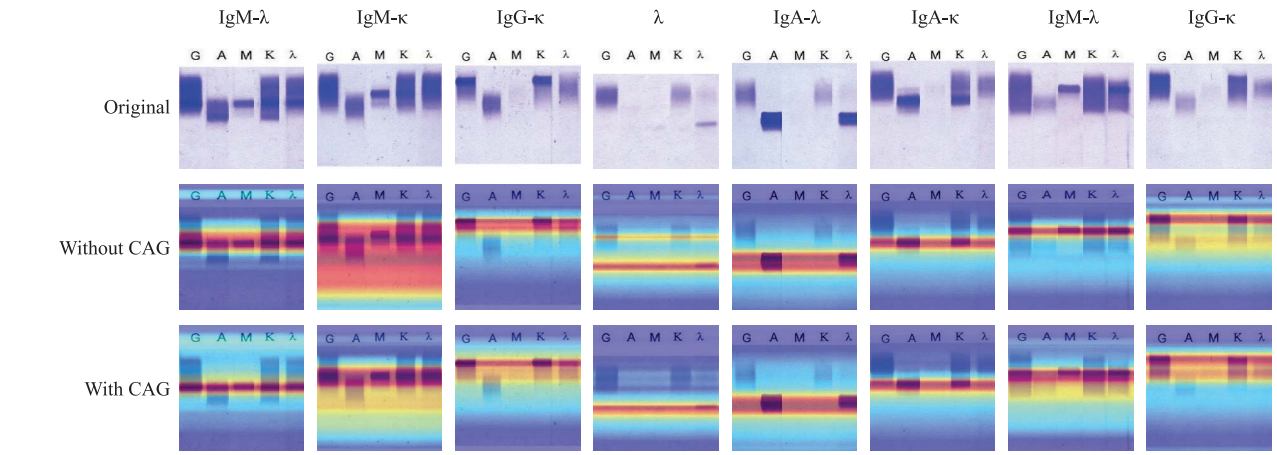

Fig. 7. Examples of saliency maps generated without and with CAGs. The better the dark-red band of a saliency map matches the target co-location, the better the result is. The $1^{st} \sim 4^{th}$ columns of the 3^{rd} row are with significant performance gain over the corresponding examples without CAGs (see saliency maps at the 2^{nd} row), because the multiple dark-red bands have converged into a single band which matches the target co-location well. In this respect, The $5^{th} \sim 6^{th}$ columns of the 3^{rd} row are with minor performance gain, while the $7^{th} \sim 8^{th}$ are with performance degradation.

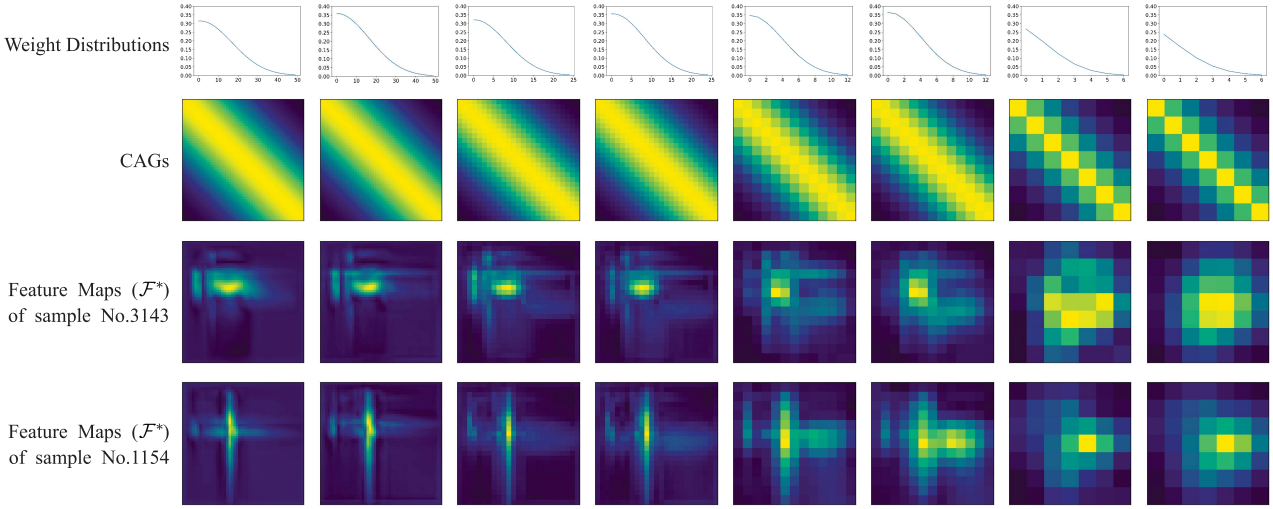

Fig. 8. CAGs weight distributions and masks at the $1^{st} \sim 8^{th}$ blocks (from left to right) of ResNet18, and the resulting feature maps ($\mathcal{F}^*$) of samples No.3143 and No.1154 after CAGs regulation. It is easy to see from the feature maps of the 3^{rd} and 4^{th} rows that, with the adaptive CAG regulation, the attention is converging from multiple regions (for early stage blocks) to a single region (for early stage blocks). This is an indication that the inference has been successfully conducted which eventually locates the correct co-location pattern from multiple candidates at early stages.

are decreasing from values around 0.35 to 0.25 along the feed-forward direction. Given the fact that all deep networks are shifting their focus (during forwarding) from low-level features (i.e., specific bands in this study) to high-level patterns (i.e., attention regions), the weight decreasing is a reasonable resulting strategy that coaches the inference to rely less on the CAGs when attention regions are more certain at later blocks. The resulting feature maps (i.e., $\mathcal{F}^*$ after CAG filtering) of the 2 samples in Fig. 8 has further confirmed this effect, in which the learning has successfully converged to a single peak (corresponding to an attention region) at later blocks from the multiple peaks at the early blocks.

E. Comparison to the State-of-the-Art

To further investigate the performance of the proposed collocative learning framework, we compare it to ResNet18 and the other 3 state-of-the-art (SOTA) methods including

- DenseNet [43] which has been considered a logical extension of the popular ResNet framework but is with fewer parameters and comparable performance;
- CBAM [37] which is one of the most popular attention mechanism that is open to any base models (we use ResNet18 in this study);
- Inception-v3 [14] which is a representative of complex deep networks pre-trained with a very large dataset (over 1 million images of ImageNet [32]) and with promising performance.

The result of performance comparison is shown in Table IV. Our method shows a comparable F1-score with while a dominating superiority over the start-of-the-art methods with performance gain of $61.92\% \pm 4.56$ in IoU, $76.04\% \pm 4.58$ in Precision^R, and $72.94\% \pm 4.53$ in Recall^R. Its performance gain over the base model ResNet has even reached 741.30% in IoU, 118.38% in Precision^R, and 725.11% in Recall^R. It is

TABLE IV

COMPARISON OF ACCURACY AND EFFICIENCY WITH THE STATE-OF-THE-ART. THE BEST RESULTS ARE IN BOLD. THRESHOLDS FOR CALCULATING THE IOUS OF RESNET, DENSENET, CBAM, AND INCEPTION-V3 ARE FINE-TUNED FOR BEST PERFORMANCE

Model	F1-score (%)	IoU (%)	Precision^R (%)	Recall^R (%)	Forward (ms)	Backward (ms)
ResNet18 [8]	92.09±1.84	7.36±2.80	34.82±10.06	8.84±3.61	**60.81±3.67**	**167.33±8.53**
DenseNet [43]	**94.84±1.83**	14.27±1.59	17.79±1.75	43.81±5.87	138.30±7.05	365.40±17.90
CBAM [37]	93.81±0.54	15.26±1.10	20.90±0.40	40.94±6.95	66.53±3.71	178.64±8.06
Inception-v3 [14]	94.42±3.30	3.90±2.69	15.05±10.46	5.05±3.31	107.06±5.23	279.43±11.95
Ours	94.20±1.37	**61.92±4.56**	**76.04±4.58**	**72.94±4.53**	141.67±5.65	232.33±8.03

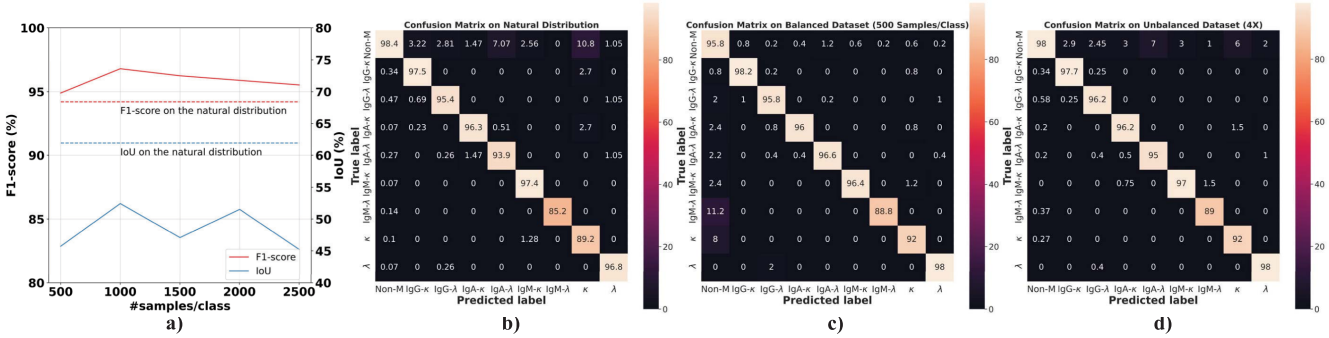

Fig. 9. Performance comparison on the balanced datasets and selected confusion matrices (numbers are in percentage): a) results on the balanced dataset, b) confusion matrix of the model trained on the datasets with the natural distribution (a real but imbalanced distribution), c) confusion matrix obtained on the balanced distribution of $\#samples/class = 500$, and d) confusion matrix obtained on the imbalanced distribution $4X$. In terms of F1-score, minor improvement has been observed on the balanced dataset compared to that on the imbalanced dataset. The improvement is mainly from the minority classes (e.g., IgM-λ). In terms of IoU, the performance drops significantly when the dataset is balanced.

interesting that the CBAM, as the representative for attention mechanisms, has obtained the best performance in IoU among the state-of-the-art methods compared (i.e., outperforms the other 3 methods by 6.94% ∼ 107.34%), which again confirms the importance of considering attention in IFE recognition. In this regard, our method has provided a formulated way of integrating attention using CAGs and thus has obtained better performance over CBAM by 305.77% in IoU, 263.83% in Precision^R, and 78.16% in Recall^R.

F. Imbalance and Noise Issues of the Dataset

In the experiments above, the IFE dataset has followed the natural distribution of the IFE sampling. However, it is not a balanced distribution in a machine learning perspective. In this section, we investigate how this will affect the learning. In addition, we will study the effects of the noise as another common issue that one many consider when starting the learning.

1) Performance on Balanced Datasets: By fixing the number of samples in each class, we can generate a balanced dataset by under-sampling the majority classes and duplicating samples in the minority classes using the original imbalanced IFE dataset.[4] In this section, we generate a set of such balanced datasets for comparison by varying the number of samples per class to 500, $1K$, $1.5K$, $2K$, and $2.5K$, respectively. The large the number is, there are more duplicated samples in each class. The performance of the proposed method on these balanced datasets are shown in Fig. 9(a).

By comparing with that of the imbalanced dataset, the performance has been improved by 0.74% ∼ 2.76% in terms of F1-score when the classes are balanced. It is not surprising that the improvement is mainly contributed by the minority classes (e.g., IgM-λ) whose original class sizes are smaller than 100. This can be observed more clearly from the confusion matrices in Fig. 9, where the false negative rate of IgM-λ has been reduced by 12.29% ∼ 57.39% on the balanced datasets. This is simply because the impact of the minority classes in the resulting models has been increased when more samples from them have been fed into the networks.

By contrast, in terms of IoU, the performance has decreased, which is less intuitive as in most of the learning problems in which we can obtain better performance when classes are balanced. However, by reviewing the samples on which the balanced models have inferior IoU performance, we found most of these samples are with sharp target band edges aside to several other suspicious bands as distractions (e.g., the samples with degraded IoU performance in Fig. 10). While the predicted labels are all correct, the models have put the reference centers on the band edges rather on the band centers. This is counter-intuitive but is not wrong indeed, because edges are more noticeable than the centers in the area with smooth color change. This type of inference is particularly effective when there are other distraction bands, because it depends on less references. The models may have developed such "tricks" when the portion of these samples have been increased in the balanced datasets. This is an advanced ability for IFE recognition but unfortunately not IoU friendly. Another evidence, which indicates that, by balancing the classes, the models

[4]Note that methods like SMOTE are commonly used for generating balanced datasets. However, these methods are not able to generate co-location references because they operate in the feature space rather than generating actual samples. We are using the under- and over-sampling in the experiments, so that the new samples can inherit the co-location reference and the IoU can be measured.

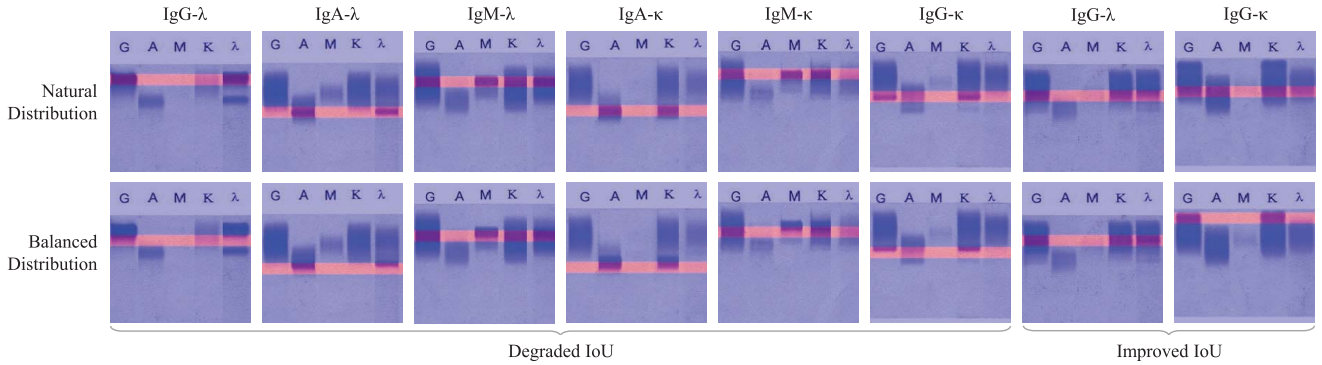

Fig. 10. Typical results on samples with multiple suspicious co-location bands before (the 1^{st} row) and after (the 2^{nd} row) balancing the class distribution. These with degraded IoU performance are resulted from the a new skill the model have developed that recognizes a protein type by simply referring to the cutting edges of the target bands (instead of the bands themselves). This is an useful skill which utilizes less information to avoid the distractions of the other suspicious co-location bands, but unfortunately not human intuitive and IoU friendly. This would not happen when the target band edges are smoother and there are no such edges for references (see examples with improved IoU performance).

can develop better ability of recognizing IFE co-locations, is that there are a significant amount of samples on which the balanced models have obtained better IoU performance than that of the imbalanced models. Those samples are also with multiple distractions but without cutting band edges (Fig. 10). The balanced models have successfully skipped the distractions and located the target bands on which the imbalanced models have failed. However, the ratio between the multiple-distraction samples with and without cutting target band edges is roughly 3 to 1, which causes the degrade of the overall IoU performance.

2) What Will Happen When the Dataset Scales Up?: Practically, given the aforementioned observations, one may have the concern that whether the performance will be affected when more samples are collected in the future. Since we can observe from Fig. 9 that it would not affect the performance much when the classes are balanced, in this section, we study what will happen if datasets followed a natural (imbalanced) distribution scale up.

We have simulated this situation by conducting the experiments on 4 datasets (denoted as $1\times$, $2\times$, $3\times$, and $4\times$ hereafter) that are with the sizes of 1, 2, 3, and 4 times larger than that of the original dataset respectively. These datasets are generated by randomly selecting and duplicating samples from the original dataset. To determine the number of samples of a class in the dataset kX (with the size of k times larger than that of the original one), we first round up its original number of samples (i.e., $\#samples$ in Table I) to the nearest hundred and then multiple the result with k. For example, the $\#samples$ of IgM-λ will be 3×100 in the dataset $3X$ given its original $\#samples$ 27. This will keep the original distribution when the dataset size grows.

The results on these 4 datasets are given in Table V, where we have obtained almost the same performance as that on the original dataset. The performance in both F1-score and IoU are stable enough and even with slight improvement in F1-score. By combining with the observations in Section IV-F.1, we can see that whether the models will develop the counter-intuitive skills is mainly determined by the class distribution rather than the size of the datasets. It is easy to understand because the human technicians' way of recognizing the protein types has

TABLE V

PERFORMANCE COMPARISON OF MODELS TRAINED ON THE ORIGINAL (REAL) DATASET AND SYNTHETIC DATASETS WITH SIZE OF 1, 2, 3, AND 4 TIMES ($1\times$, $2\times$, $3\times$, AND $4\times$) LARGER THAN THE ORIGINAL DATASET. THE SYNTHETIC DATASETS ARE USED TO SIMULATE THE SITUATION THAT A DATASET FOLLOWING NATURAL DISTRIBUTION SCALES UP. THE BEST PERFORMANCE ARE IN BOLD

Dataset	F1-score (%)	IoU (%)	Precision^R (%)	Recall^R (%)
Real	94.20±1.37	**61.92±4.56**	**76.04±4.58**	**72.94±4.53**
$1X$	94.95±2.54	57.15±10.16	70.79±10.85	68.14±10.35
$2X$	95.42±1.26	56.88±5.58	71.11±5.63	68.49±5.16
$3X$	95.31±2.37	56.21±7.37	70.43±7.41	68.00±7.15
$4X$	**95.55±2.52**	58.43±4.82	72.61±4.98	70.26±4.78

also been developed on the natural distribution rather than a balanced one. By keeping the natural distribution for learning, we can have the machine inference to be more consistent to the human experience.

3) The Noise Issues: One may have noticed from the IFEs in the figures above that there are various types of noise. Will these noise affect the performance? Should we conduct preprocessing before we start the learning? These are common questions that one may raise in image analysis tasks. To answer these questions, we generate another set of datasets by adding noise to the samples randomly selected from the original dataset. There are 4 types of noise, namely salt, pepper, white, and Gaussian blur from which we will randomly select one for a new sample. To simulate the natural distribution in a real scenario, the datasets are with an imbalanced class distribution by rounding up its original numbers of samples to the nearest hundreds. The datasets are different from each other in the noise ratio which are selected from the range $[0, 0.9]$ with a step 0.1.[5] The results are shown in Fig. 11, where we can see the performance is insensitive to the noise when the ratio is in the range of $[0, 0.30]$. This is an encouraging result, because human technicians usually will discard the noise samples when the ratio is larger than 0.15.

[5]In the experiment, we have controlled the noise ratio only for the other 3 types of noise except the Gaussian blur. For the Gaussian blur, we fixed the ratio to 0.01 because it controls the degree of blur, but in a real laboratory setting, the IFE machines are all with auto-focus camera and thus rarely generate blur images.

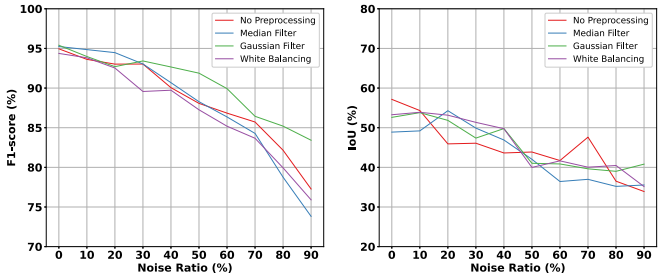

Fig. 11. Performance comparison of using different preprocessing methods on the datasets generated by varying the noise ratio. The performance are not affected much by the noise in the human acceptable range of noise ratio [0, 0.15], and the preprocessing has provided limited improvement to the performance.

To verify whether the noise reduction techniques work, we further repeat the experiment by adding the Median, Gaussian filtering, and white balancing for preprocessing respectively. The results are shown in Fig. 11, where we can see none of the preprocessing has brought significant improvement to the collocative learning.

It is worth mentioning that the investigations above are all based on global noise assumption, because this is the most possible noise type in a real laboratory setting where the samples with local noise will be discarded. Furthermore, it is easy to understand that the collocative learning is insensitive to global noise because the collocative tensor is based on the correlations of strips. As long as two strips are with the same type of global noise, their correlation would not change much.

V. Conclusion

We have proposed a collocative learning framework for IFE analysis in which we have constructed a collocative tensor to transform the binary relation of target bands into unary relation in the tensor so that the sophisticated deep learning models can be employed. With the tensor, we have proposed a label-free-location method which eliminates the requirements of location labels while providing the exact localization of attention regions. A network called Coached CAGs has also been proposed to guide the inference to be more consistent with human logic. The performance of the framework has been validated in the experiments.

While encouraging results have been observed, there is still space for each of the components in the framework to be optimized. For example, in the current form of Eqn. (2), the three factors, namely the $\Phi^p(s_i, s_j)$, $\Psi^p(s_i, s_j)$, and $\pi^p(s_i, s_j)$, are with equal effect on the result. It is the most intuitive and straightforward fusion schema one may consider as an initial attempt. However, it might not be an optimal solution. Besides, we have observed the models have learned advanced but not intuitive skills (i.e., recognizing proteins from cutting band edges) by balancing the classes. It is worth future study to make the reasoning skills consistent to human logic.

References

[1] D. Pulte *et al.*, "Trends in survival of multiple myeloma patients in Germany and the United States in the first decade of the 21st century," *Brit. J. Haematol.*, vol. 171, no. 2, pp. 189–196, Oct. 2015.

[2] E. Terpos, "Multiple myeloma: Clinical updates from the American society of hematology annual meeting, 2017," *Clin. Lymphoma Myeloma Leukemia*, vol. 18, no. 5, pp. 321–334, May 2018.

[3] R. L. Siegel, K. D. Miller, and A. Jemal, "Cancer statistics, 2019," *CA: A Cancer J. Clinicians*, vol. 69, no. 1, pp. 7–34, 2019.

[4] B. H. Menze *et al.*, "The multimodal brain tumor image segmentation benchmark (BRATS)," *IEEE Trans. Med. Imag.*, vol. 34, no. 10, pp. 1993–2024, Oct. 2015.

[5] S. G. Armato, M. L. Giger, and H. Macmahon, "Automated detection of lung nodules in CT scans: Preliminary results," *Med. Phys.*, vol. 28, no. 8, pp. 1552–1561, Aug. 2001.

[6] H. D. Cheng, J. Shan, W. Ju, Y. Guo, and L. Zhang, "Automated breast cancer detection and classification using ultrasound images: A survey," *Pattern Recognit.*, vol. 43, no. 1, pp. 299–317, Jan. 2010.

[7] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[9] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[10] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2961–2969.

[11] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[12] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2015, pp. 234–241.

[13] G. Litjens *et al.*, "A survey on deep learning in medical image analysis," *Med. Image Anal.*, vol. 42, pp. 60–88, Dec. 2017.

[14] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.

[15] K.-K. Maninis, J. Pont-Tuset, P. Arbeláez, and L. Van Gool, "Deep retinal image understanding," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2016, pp. 140–148.

[16] A. Jamaludin, T. Kadir, and A. Zisserman, "SpineNet: Automatically pinpointing classification evidence in spinal MRIs," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2016, pp. 166–175.

[17] S. S. Han, M. S. Kim, W. Lim, G. H. Park, I. Park, and S. E. Chang, "Classification of the clinical images for benign and malignant cutaneous tumors using a deep learning algorithm," *J. Investigative Dermatol.*, vol. 138, no. 7, pp. 1529–1538, 2018.

[18] O. Oktay *et al.*, "Multi-input cardiac image super-resolution using convolutional neural networks," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2016, pp. 246–254.

[19] V. Gulshan *et al.*, "Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs," *J. Amer. Med. Assoc.*, vol. 316, no. 22, pp. 2402–2410, 2016.

[20] A. Esteva *et al.*, "Dermatologist-level classification of skin cancer with deep neural networks," *Nature*, vol. 542, no. 7639, pp. 115–118, Feb. 2017.

[21] R. Sa *et al.*, "Intervertebral disc detection in X-ray images using faster R-CNN," in *Proc. 39th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2017, pp. 564–567.

[22] J. Ding, A. Li, Z. Hu, and L. Wang, "Accurate pulmonary nodule detection in computed tomography images using deep convolutional neural networks," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2017, pp. 559–567.

[23] M. Liu, J. Dong, X. Dong, H. Yu, and L. Qi, "Segmentation of lung nodule in CT images based on mask R-CNN," in *Proc. 9th Int. Conf. Awareness Sci. Technol. (iCAST)*, Sep. 2018, pp. 1–6.

[24] J. Liu and P. Li, "A mask R-CNN model with improved region proposal network for medical ultrasound image," in *Proc. Int. Conf. Intell. Comput.* Cham, Switzerland: Springer, 2018, pp. 26–33.

[25] M. A. Al-masni *et al.*, "Simultaneous detection and classification of breast masses in digital mammograms via a deep learning YOLO-based CAD system," *Comput. Methods Programs Biomed.*, vol. 157, pp. 85–94, Apr. 2018.

[26] J. George *et al.*, "Using YOLO based deep learning network for real time detection and localization of lung nodules from low dose CT scans," *Proc. SPIE*, vol. 10575, Feb. 2018, Art. no. 105751I.
[27] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3D U-net: Learning dense volumetric segmentation from sparse annotation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2016, pp. 424–432.
[28] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: A nested U-net architecture for medical image segmentation," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*. Cham, Switzerland: Springer, 2018, pp. 3–11.
[29] O. Oktay *et al.*, "Attention U-net: Learning where to look for the pancreas," 2018, *arXiv:1804.03999*. [Online]. Available: http://arxiv.org/abs/1804.03999
[30] Z. Wang, N. Zou, D. Shen, and S. Ji, "Non-local U-Nets for biomedical image segmentation," in *Proc. AAAI*, 2020, pp. 6315–6322.
[31] S. Kevin Zhou *et al.*, "A review of deep learning in medical imaging: Imaging traits, technology trends, case studies with progress highlights, and future promises," 2020, *arXiv:2008.09104*. [Online]. Available: http://arxiv.org/abs/2008.09104
[32] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
[33] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 740–755.
[34] A. Krizhevsky *et al.*, "Learning multiple layers of features from tiny images," Dept. Comput. Sci., Univ. Toronto, Toronto, ON, Canada, Tech. Rep., 2009.
[35] 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 *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 618–626.
[36] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.
[37] S. Woo, J. Park, J.-Y. Lee, and I. So Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19.
[38] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7794–7803.
[39] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-CAM++: Generalized gradient-based visual explanations for deep convolutional networks," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 839–847.
[40] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2921–2929.
[41] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 26, no. 1, pp. 43–49, Feb. 1978.
[42] M. Müller, "Dynamic time warping," in *Information Retrieval for Music and Motion*. Berlin, Germany: Springer, 2007, pp. 69–84.
[43] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4700–4708.