Refining Prototypes with Selective Feature Maps: A Lightweight Approach to Few-Shot Medical Image Classification

Ranjana Roy Chowdhury
Department of Computer Science & Engineering Indian Institute of Technology Ropar

Deepti R. Bathula
Department of Computer Science & Engineering Indian Institute of Technology Ropar

ABSTRACT
Prototypical Networks operate by embedding both support and query samples into a common feature space and then represent-ing each class with the mean vector of its support embeddings. Yet, the inherent complexity of medical imagery pose significant challenges for isolating features that are both precise and depend-able. Consequently, constructing effective prototypes in this do-main demands not only sophisticated preprocessing and more pow-erful embedding architectures, but also deliberate refinement of feature representations. In this context, most important and rep-resentative feature map selection is critical. We introduce Selec-tive Feature Representation in Prototypical Networks, a lightweight yet effective enhancement to prototype-based few-shot learning. Proposed approach explicitly refines support embeddings by rank-ing and selecting the top feature maps for each class, leveraging an ensemble of channel-wise statistics—Global Average Pooling, Max Pooling, and Variance. Built on a compact CONV4 back-bone, proposed method outperforms much larger state-of-the-art models on two medical benchmarks: achieving 67.18% (1-shot) and 78.20% (5-shot) on Derm7pt skin-lesion classification, and 63.39% (1-shot), 77.17% (5-shot), and 83.06% (10-shot) on Blood-MNIST pathology classification. These gains demonstrate that tar-geted feature-map selection significantly improves prototype qual-ity and generalization with minimal complexity, offering a practical solution for resource-constrained clinical applications.

General Terms
Meta Learning, Few Shot Learning, Prototypical Networks

Keywords
Meta Learning, Few Shot Learning, Prototypical Networks, Feature Map Selection

1. INTRODUCTION
Since the past few years, deep learning has brought a revolution in diverse fields of mankind through its astonishing features and extremely powerful prediction capabilities throughout various dis-ciplines [12, 13]. It has brought great prosperity in many industries like finance, automotive, manufacturing including healthcare [15] e.g., computer-aided diagnosis [5], medical image retrieval [17], medical image mining [2], medical image sorting [23], etc. Deep learning and its applications have had a huge impact on the health-care industry and have uplifted diagnostic capability by their out-standing features, flexibility, and robustness. But the main draw-back with these models lies in their hunger for huge annotated samples to achieve efficient performance. Researchers have ex-plored various techniques to address this challenge, including trans-fer learning [26] and domain adaptation [8], which leverage large source domains to provide models with rich feature representations and favorable initialization for generalization. However, these ap-proaches often struggle in extreme scenarios, such as severely tail-distributed data or significant domain shift.

The necessity for large numbers of labeled samples poses a huge challenge in data-scarce domains such as medicine. Even though many approaches have been proposed, learning from very few an-notated samples remains an open challenge. This contrasts with hu-man intelligence, which adapts quickly in a new environment using prior knowledge. This is where meta-learning comes in—it lever-ages exactly this human-like ability to learn and react from very few examples [10]. This is the crux of meta-learning, which repli-cates learning from very few samples. Recently, meta-learning has gained popularity as an alternative in low-data regimes, especially in medicine. For instance, in predicting extremely rare disease classes with almost no samples, meta-learning can be highly sig-nificant. This few-sample prediction scenario is termed Few-Shot Learning [25]. Various meta-learning techniques have been ap-plied to Few-Shot problems: metric-based approaches like Match-ing Networks [24] and Prototypical Networks [22], gradient-based methods such as MAML [7] and Optimization-Based Few-Shot [20].

Prototypical Networks [22] have become popular owing to their simple and intuitive design where they form class prototypes by av-eraging support-set embeddings and then classify queries according to the nearest-prototype distance. However, this vanilla formula-tion treats every support embedding equally, which may not reflect their true relevance or importance in class prototype formation. Several studies [9, 21] have introduced support-sample reweight-ing schemes to improve prototype construction, including the pre-

viously proposed work IPNET [18] that employed Maximum Mean Discrepancy to identify and emphasize the most influential examples during prototype formation. In parallel, methods such as COMET [3] have explored in extracting semantically meaningful "Concepts" from each class sample to build "Concept-Based Class Prototypes".

Feature selection plays a critical role in enhancing the discrimi-native power and robustness of deep representations, particularly in high-dimensional settings like few-shot learning. Recent studies have highlighted that not all features contribute equally to the fi-nal prediction, and the presence of redundant or noisy dimensions can adversely affect the learning process [14, 27]. Works such as

[4] and [11] have shown that adaptive filtering or attention-based weighting of features can significantly improve generalization and reduce overfitting, especially in low-data regimes. In the context of prototype-based few-shot methods, this becomes even more per-tinent since prototype computation depends heavily on the aggre-gated feature embeddings.

Despite these advances, none of the existing approaches explic-itly emphasize filtering out noisy or less informative feature maps within each embedding; they still aggregate all feature maps when computing prototypes, potentially diluting the representation with irrelevant information. In this work, we propose Selective Feature Representation in Prototypical Networks which is a novel feature selection strategy that retains only the most salient feature maps in support sample embeddings, thereby reducing noise and enhanc-ing class prototype quality. Proposed method determines important feature maps through an ensemble of three techniques—Global Av-erage Pooling, Max Pooling, and Variance ensuring both diversity and robustness in the resulting class prototypes. The major contri-butions of this paper are as follows:

—This work proposes refining prototypes in prototypical networks through selective feature maps, where only the most influential support embedding feature maps are retained for class prototype formation.

—The feature maps are selected using an ensemble of three techniques—Global Average Pooling, Max Pooling, and Variance—ensuring a diverse and informative representation by capturing complementary aspects of the feature space.

—The effectiveness of the proposed method is demonstrated on two medical imaging domains: dermatoscopic imaging using the Derm7pt [6] dataset and pathology using the BloodMNIST [1] dataset.

—t-SNE visualizations on the BloodMNIST dataset show that Selective Feature Representation in Prototypical Networks yields more compact intra-class clusters and clearer inter-class separa-tions than vanilla Prototypical Networks.

The paper begins by introducing the preliminaries in Section 2, then describes the proposed methodology in Section 3. Section 4 details the experimental setup and analysis, followed by the presentation of results in Section 5. Finally, Section 6 concludes the paper and outlines directions for future work.

2. PRELIMINARIE

S 2.1 Meta Learning

Meta-learning, or "learning to learn" [7], aims to train models that can swiftly master new tasks from only a few examples by optimizing for adaptability rather than a single fixed objective. Given a labeled dataset D with C classes, we first partition it into

two non-overlapping subsets, $D_{Meta\text{-}Train}$ and $D_{Meta\text{-}Test}$, ensur-ing that classes seen during meta-training do not appear in meta-

testing. During the meta-training phase, over E_{train} episodes we sample a task T_i from $D_{Meta\text{-}Train}$ and split it into a small sup-port set S_i (used to update the meta-learner f_θ via gradient descent

on the loss $L(f_\theta(S_i), y_{S_i})$ and a query set Q_i (used to evaluate $L(f_\theta(Q_i), y_{Q_i})$, thereby refining the model's few-shot learning strategy through episodic training. In the meta-testing phase, we as-sess generalization over E_{test} episodes by sampling tasks T_j from

$D_{Meta\text{-}Test}$, fine-tuning f_θ on each support set, and measuring per-formance on the corresponding query set; the aggregate query loss

across these episodes quantifies the model's ability to adapt to entirely new tasks with minimal data.

2.2 Few Shot Learning

Conventional deep learning approaches typically depend on large volumes of labeled data, which becomes a significant limita-tion in domains such as medical imaging, where data imbalance and scarcity—especially for rare classes—are common. Few-Shot Learning (FSL) [20] offers an effective alternative by enabling models to perform classification tasks using only a small number of labeled instances.

In Few-Shot Learning, tasks are framed as N-way K-shot classifi-cation problems, where N indicates the number of distinct classes involved in a given task and K specifies the number of available support samples per class. This episodic setting is designed to sim-ulate the conditions under which the model will be evaluated. As described in Algorithm 1, the meta-training phase involves generat-

ing E_{Train} such tasks T_i, each structured according to the N-way K-shot format. An analogous process is followed during meta-testing,

where the model is assessed on E_{Test} previously unseen tasks drawn from a separate test distribution.

2.3 Prototypical Networks

Meta-learning approaches have become key to addressing Few-Shot Learning (FSL) challenges especially in data scarce fields like medical imaging—by enabling rapid adaptation from limited examples. Among these, metric-based strategies [16] ex-cel in both simplicity and accuracy; notably, Prototypical Networks

[22] perform N-way K-shot classification by embedding support set examples, computing each class prototype as the mean of its feature vectors, and assigning queries to the nearest prototype in embedding space.

Formally, given a labeled dataset $D = \{(x_k, y_k)\}^n_{k=1}$ drawn from class set C, we split C into disjoint training and testing subsets C_{tr} and C_{ts} (so that $C_{tr} \cap C_{ts} = \emptyset$), thereby ensuring that during evalua-tion the model must generalize to entirely unseen classes using only the few support examples provided. The training procedure follows a meta-learning framework consisting of repeated episodes. Dur-ing each meta-training episode, a task T_i is created by randomly sampling N classes from C_{tr}. For each class c within the task, K samples form the support set S_i^c, and Q samples form the query set Q^c_i, both drawn from $D_{Meta\text{-}Train}$.

A class prototype p^c_i for each class c is computed by averaging the feature representations of its support set:

$$p_i^c = \frac{1}{K} \sum_{x_k \in S_i^c} f_\theta(x_k) \quad (1)$$

For each query sample $(x_q, y_q) \in Q^c_i$, the distance to each proto-type is calculated. The probability that the query belongs to class c

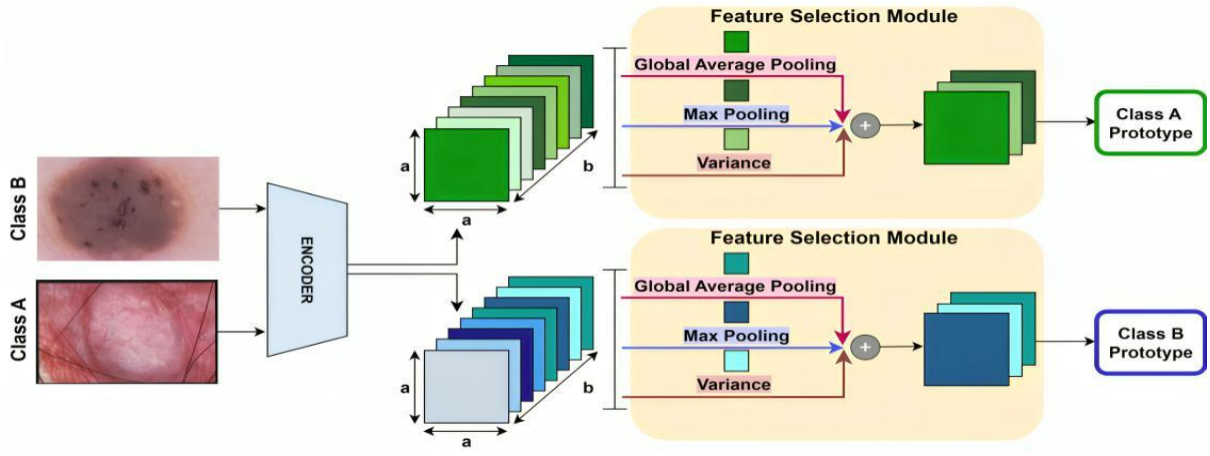


Fig. 1: Overview of the proposed refinement of prototypes in Prototypical Networks, where feature maps from each support embedding are selected using an ensemble strategy that combines global average pooling, max pooling, and variance-based selection.

is obtained by applying a softmax over the negative distances:

$$p_\theta(y = c \mid x_q) = \frac{\exp(-d(f_\theta(x_q), p_i^c))}{\sum_{c'} \exp(-d(f_\theta(x_q), p_i^{c'}))} \quad (2)$$

Here, $d(f_\theta(x_q), p_i^c)$ denotes the distance between the query embedding $f_\theta(x_q)$ and the prototype p_i^c. The model is optimized by minimizing the negative log-likelihood of the true class label over all query samples, thereby updating the parameters θ of the embedding function f_θ.

In the meta-testing phase, the trained model is evaluated on new tasks sampled from C_{ts}, which contain previously unseen classes. This phase serves to assess how well the model generalizes beyond the training distribution.

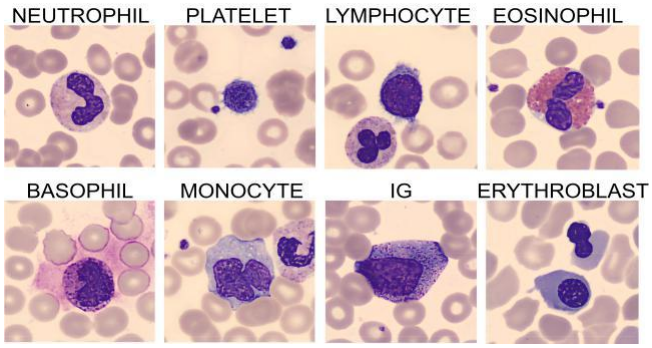


Fig. 2: Sample images from random classes of BloodMNIST dataset

3. PROPOSED METHODOLOGY

The core objective of this study is to eliminate redundant or noisy feature maps and preserve only those that are most informative, thereby enabling the construction of more robust class prototypes. We posit that not all extracted feature maps contribute positively to prototype formation—some may introduce extraneous informa-tion that could impair predictive performance. Consider an input

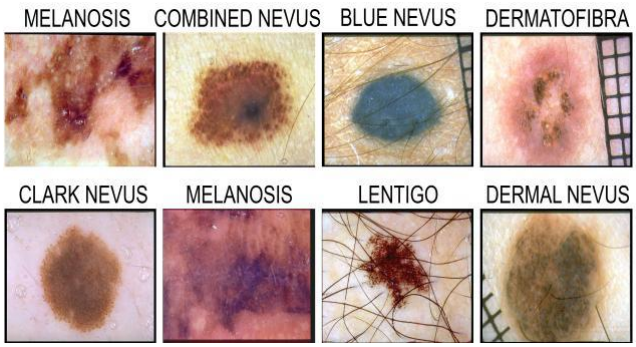


Fig. 3: Sample images from random classes of Derm7pt dataset

image $x \in R^{A\times A\times B}$, where A × A denotes the spatial resolu-tion and B the number of input channels. When processed by the encoder $f_\theta : R^{A\times A\times B} \rightarrow R^{a\times a\times b}$, a pre-final feature tensor $z = f_\theta(x) \in R^{a\times a\times b}$ is extracted where a×a is the reduced spatial resolution and b is the number of feature channels. From this ten-sor, we select the top M feature maps by integrating three comple-mentary selection strategies: Global Average Pooling, Max Pool-ing, and Variance analysis. Global Average Pooling, Max Pool-ing, and Variance each capture complementary aspects of feature maps, and their integration leads to richer prototype representa-tions. Global Average Pooling summarizes the overall activation strength across spatial locations, offering a sense of the dominant trends or global context within each channel. Max Pooling empha-sizes the most strongly activated regions, often corresponding to highly discriminative or distinctive visual cues that are critical for class separation. Variance, in contrast, quantifies the degree of fluc-tuation within a channel, highlighting feature maps where activa-tions exhibit strong diversity and thus capturing localized complex-ity or structural variation. When combined, these three statistical perspectives provide a holistic characterization of the support sam-ples: the average conveys representativeness, the maximum empha-sizes saliency, and the variance reflects diversity and richness of vi-sual patterns. This multi-faceted view ensures that prototypes are not only broadly representative but also sensitive to fine-grained

Algorithm 1 Meta-learning with Selective Feature Representation in Prototypical Networks

Input:
* Dataset $D = \{(x_k, y_k)\}_{k=1}^{n}$ with $y_k \in \{1, \ldots, C\}$
* C_{tr}, C_{ts}: disjoint train/test class splits ($|C_{tr}| + |C_{ts}| = C$)
* $D_{Meta\text{-}Train}$, $D_{Meta\text{-}Test}$: subsets of D for train/test classes
* E_{Train}, E_{Test}: number of meta-training/testing episodes
* N: classes per episode
* K, N_Q: support/query samples per class
* M: number of top feature-map channels per support sample to be chosen by Global Average Pooling, Max Pooling and Varianc

Randomly initialize parameters θ of Meta-Learner f

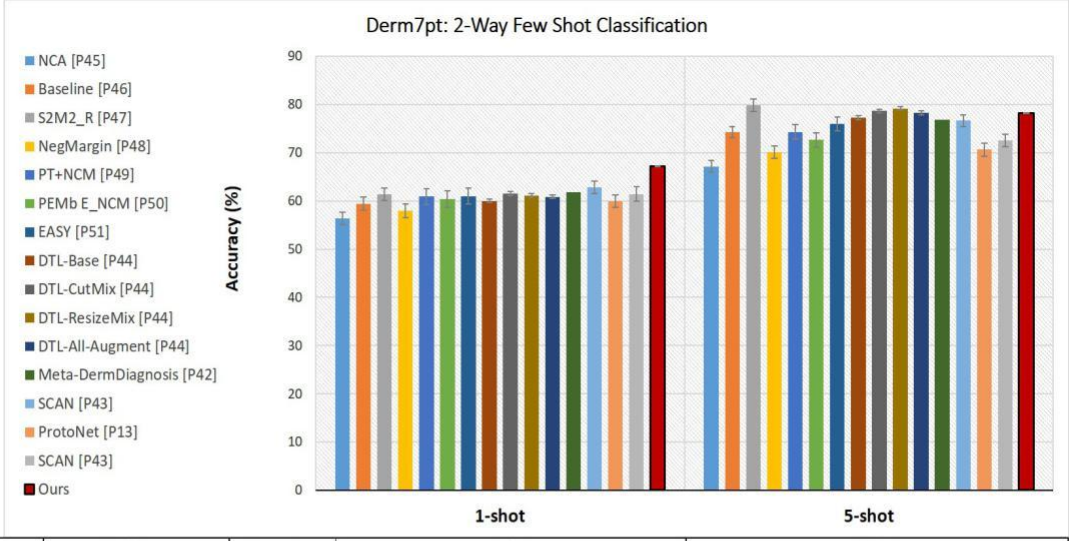
Meta-Training Phase (for $i = 1, \ldots, E_{Train}$):
  Sample N classes $C_i \subset C_{tr}$
  Construct support set S_i and query set Q_i:
  for $c \in C_i$ do
    $D_i^c \leftarrow \{(x, y) \in D_{Meta\text{-}Train} \mid y = c\}$
    $S_i^c \leftarrow$ random K samples from D_i^c ▷ Form support set pf class c
    $Q^c{}_i \leftarrow$ random N_Q samples from $D_i^c \setminus S_i^c$ ▷ Form query set pf class c
    $S_i \leftarrow S_i \cup S_i^c$, $Q_i \leftarrow Q_i \cup Q^c{}_i$ ▷ Feature-map selection for each support sample
    for $z_{c,j} = f_\theta(x_{c,j})$ in S_i^c do
      $a_{c,j} \leftarrow \text{GlobalAvgPool}(z_{c,j}) \in R^b$ ▷ Calculate Global Average Pooling across all channels ▷
      $m_{c,j} \leftarrow \text{MaxPool}(z_{c,j}) \in R^b$ Calculate Max Pooling across all channels ▷
      $v_{c,j} \leftarrow \text{Var}(z_{c,j}) \in R^b$ Calculate Varince across all channels
      Sort channels by descending $a_{c,j}$, $m_{c,j}$, $v_{c,j}$
      Interleave top indices to form $I_{c,j}$ with $|I_{c,j}| = M$
      $\hat{z}_{c,j} \leftarrow \text{Mask}(z_{c,j}, I_{c,j})$
    end for ▷ Class Prototype with masked feature map
    $p_i^c = \frac{1}{K} \sum_{j=1}^{S_i^c} \hat{z}_{c,j}$, $p_i^c \in R^b$
  end for
  $L \leftarrow 0$ ▷ Initialize Loss
  for $(x_k, y_k) \in Q_i$ do ▷ For each query sample in Q ▷ Compute prediction probability of each query
    Extract $f_\theta(x_k) \rightarrow \hat{z}_k$ using Step 18 to 25
    $p_\theta(y = c \mid \hat{z}_k) = \frac{\exp -d(\hat{z}_k, p_i^c)}{\sum_{c'} \exp -d(\hat{z}_k, p^{c'})}$
    $L \leftarrow L + \frac{1}{N*Q} [d(\hat{z}_k, p_i) + \log \sum_{c'} \exp(-d(\hat{z}_k, p_{i'}))]$ embedding ▷ Compute Cross Entropy Loss
  end for
Update $\theta \leftarrow \theta - \alpha \nabla_\theta L$ ▷ Update parameter θ of model f

Meta-Testing Phase: repeat from step 10 for E_{Test} episodes on $D_{Meta\text{-}Test}$ to evaluate the performance of f_θ

details and discriminative features qualities that are particularly es-sential for image-based tasks where subtle local variations often determine class identity. Let suppose, each support sample embed-ding from a class c is represented as a feature map of dimension $a \times a \times b$, where b denotes the number of channels and $a \times a$ is the spatial resolution. For each class $c \in \{1, \ldots, N\}$ in an N-way classification setting, we extract feature maps for K support sam-ples using the encoder f_θ. Each support sample of c^{th} class is rep-resented as $z_{c,j} \in R^{a \times a \times b}$, where $j = 1, \ldots, K$. These samples collectively form the support set S_i^c, corresponding to class c in task T_i. For each support sample $z_{c,j}$, we identify the most infor-mative feature maps by using three different channel-wise statis-tics: the average pooled vector $a_{c,j} = \text{GlobalAvgPool}(z_{c,j})$, the max pooled vector $m_{c,j} = \text{MaxPool}(z_{c,j})$, and the per-channel variance vector $v_{c,j} = \text{Var}(z_{c,j})$, all of which lie in R^b. These methods are chosen because they each highlight unique aspects of the feature activations. A detailed picture of proposed feature se-lection strategy is depicted in Figure 1. The calculated vectors from these channels are then ranked based on descending values of $a_{c,j}$, $m_{c,j}$, and $v_{c,j}$. A fixed number of top M unique channels (denoted as $I_{c,j}$, where $|I_{c,j}| = M$) are selected by interleaving the sorted indices from the three metrics. After constructing the selected in-dex set $I_{c,j}$ for a support embedding $z_{c,j} \in R^{a \times a \times b}$, we apply a binary channel-wise mask that preserves the chosen channels and suppresses the rest. Formally, let $M_{c,j} \in \{0, 1\}^b$ be the binary mask such that $M_{c,j}(k) = 1$ if $k \in I_{c,j}$ and 0 otherwise. The masked tensor retains the original shape $R^{a \times a \times b}$, which simplifies batching and downstream processing. In our experiments, this zero-ing operation (retaining selected channels and suppressing others) proved effective for preserving tensor shapes and enabling efficient batching. The resulting masked embedding is denoted as $\hat{z}_{c,j}$. The prototype for class c in task T_i is then obtained by averaging the corresponding masked support features:

$$p_i^c = \frac{1}{K} \sum_{j=1}^{S_i^c} \hat{z}_{c,j}, \quad p_i^c \in R^b. \tag{3}$$

Each class prototype $p^c{}_i$ captures the most informative channel acti-vations from its support samples based on average, maximum, and variance-based criteria.For a given query sample $(x_k, y_k) \in Q^c{}_i$, we first extract feature embedding $\hat{z}$ for each query sample us-ing the proposed feature selection strategy, and its distance to each class prototype is obtained by computing the corresponding Eu-clidean distance. The probability $p_\theta(y = c \mid \hat{z})$ that the query



SOTA	Backbone	Parameters	1-shot	5-shot
NCA [P45]	WRN-28-10	~36.5M	56.32±1.29	67.18±1.15
Baseline [P46]	WRN-28-10	~36.5M	59.43±1.34	74.28±1.14
S2M2_R [P47]	WRN-28-10	~36.5M	61.37±1.33	**79.83±1.34**
NegMargin [P48]	WRN-28-10	~36.5M	58.00±1.44	70.12±1.30
PT+NCM [P49]	WRN-28-10	~36.5M	60.92±1.68	74.33±1.48
PEMb E_NCM [P50]	WRN-28-10	~36.5M	60.40±1.72	72.63±1.48
EASY [P51]	WRN-28-10	~36.5M	61.02±1.67	75.98±1.41
DTL-Base [P44]	DenseNet121	~8.0M	60.00±0.39	77.25±0.41
DTL-CutMix [P44]	DenseNet121	~8.0M	61.57±0.41	<u>78.65±0.40</u>
DTL-ResizeMix [P44]	DenseNet121	~8.0M	61.09±0.38	79.18±0.39
DTL-All-Augment [P44]	DenseNet121	~8.0M	60.88±0.40	78.30±0.40
Meta-DermDiagnosis [P42]	Conv6	83840	61.8	76.9
SCAN [P43]	Conv6	83840	<u>62.80±1.34</u>	76.65±1.21
ProtoNet [P13]	Conv4	50688	59.98±1.28	70.62±1.30
SCAN [P43]	Conv4	50688	61.42±1.49	72.58±1.28
Ours	Conv4	50688	**67.18±0.08**	78.20±0.03

Fig. 4: Performance comparison of few-shot learning methods on the Derm7pt dataset for 2-way classification under 1-shot and 5-shot settings. Accuracy (%) is reported along with model backbones and parameter sizes.(values in bold are the highest and values in <u>underline</u> are the second highest)

sample is classified into class c is then expressed using a softmax function applied over the negative distances:

$$p_\theta(y = c \mid \hat{z}_k) = \frac{\exp - d(\hat{z}_k, p_i^c)}{\sum_{c'} \exp - d(\hat{z}_k, p_i^{c'})} \quad . \quad (4)$$

where the softmax function is applied to the negative distances be-tween the query embedding $\hat{z}_k$ and the class prototypes p^c_i. Here, as mentioned in Section 2.3 also, $d(\hat{z}_k, p^c_i)$ denotes the distance be-tween the feature representation of the query sample $\hat{z}_k$ and the cor-responding class prototype p^c_i. The prototypical network is trained by minimizing the negative log-likelihood of the true class for each query sample, thereby learning the parameters θ of the feature ex-tractor f_θ. A step-by-step overview of proposed approach is out-lined in Algorithm 1.

4. EXPERIMENTATION DETAILS

4.1 Datasets

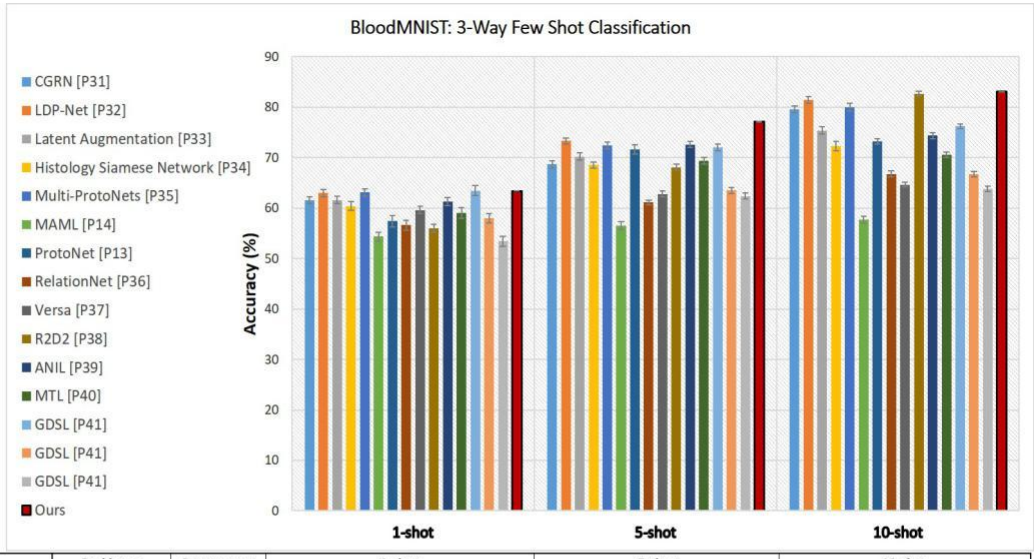
To ensure broad applicability, the proposed method was evaluated on two diverse medical imaging benchmarks, covering dermatol-ogy and pathology, thereby offering a comprehensive evaluation of its performance. The details of dataset used and its configuration are provided in Table 1. Some sample images from random classes of each dataset are shown in Figures 2 and 3 as a reference.

Table 1. : Configuration details of datasets used in this study

Dataset Summary				
Dataset	#Images	Resolution	#Train Classes	#Test Classes
Derm7pt[6]	2,000	224×224	13	7
BloodMNIST[1]	17092	84×84	4	4

4.2 Implementation Details

The experiments utilize a CONV4 backbone consisting of four se-quential blocks, each containing 64 filters of size 2 × 2, followed by ReLU activations, batch normalization, and 2 × 2 max-pooling with stride 2. We train the network with Stochastic Gradient De-scent (SGD), setting the learning rate to 0.1 and the momentum to 0.9, under a standard few-shot learning framework. All experiments were carried out on an NVIDIA A100-SXM4 GPU with 40 GB of



SOTA	Backbone	Parameters	1-shot	5-shot	10-shot
CGRN [P31]	Resnet-50	~23.9M	61.59 ± 0.67	68.67 ± 0.74	79.58 ± 0.69
LDP-Net [P32]	Resnet-50	~23.9M	62.97 ± 0.74	<u>73.31 ± 0.59</u>	81.45 ± 0.67
Latent Augmentation [P33]	Resnet-50	~23.9M	61.58 ± 0.71	70.22 ± 0.71	75.40 ± 0.78
Histology Siamese Network [P34]	Resnet-50	~23.9M	60.42 ± 0.83	68.56 ± 0.59	72.29 ± 0.91
Multi-ProtoNets [P35]	Resnet-50	~23.9M	63.10 ± 0.75	72.44 ± 0.62	80.01 ± 0.76
MAML [P14]	Resnet-25	~15–17M	54.37 ± 0.89	56.54 ± 0.75	57.69 ± 0.66
ProtoNet [P13]	Resnet-25	~15–17M	57.40 ± 1.16	71.62 ± 0.95	73.23 ± 0.55
RelationNet [P36]	Resnet-25	~15–17M	56.62 ± 0.99	61.14 ± 0.48	66.79 ± 0.66
Versa [P37]	Resnet-25	~15–17M	59.63 ± 0.79	62.79 ± 0.53	64.63 ± 0.49
R2D2 [P38]	Resnet-25	~15–17M	55.93 ± 0.79	68.06 ± 0.60	<u>82.66 ± 0.47</u>
ANIL [P39]	Resnet-25	~15–17M	61.28 ± 0.81	72.62 ± 0.61	74.36 ± 0.64
MTL [P40]	Resnet-25	~15–17M	59.07 ± 1.04	69.36 ± 0.65	70.55 ± 0.55
GDSL [P41]	Resnet-25	~15–17M	**63.47 ± 0.94**	71.97 ± 0.66	76.21 ± 0.50
GDSL [P41]	Resnet-18	~11.7M	57.99 ± 0.97	63.50 ± 0.58	66.75 ± 0.57
GDSL [P41]	CONV4	50688	53.41 ± 0.97	62.39 ± 0.59	63.81 ± 0.54
Ours	CONV4	50688	<u>63.39 ± 0.04</u>	**77.17 ± 0.05**	**83.06 ± 0.02**

Fig. 5: Performance comparison of few-shot learning methods on the BloodMNIST dataset for 3-way classification under 1-shot, 5-shot, and 10-shot settings. Accuracy (%) is reported along with model backbones and parameter sizes.(values in bold are the highest and values in <u>underline</u> are the second highest)

memory. The implementation was done in Python 3.10.4 using the PyTorch framework.

4.3 Evaluation

Few-shot learning (FSL) performance is evaluated using classifi-cation accuracy. Meta-training consists of 2000 episodes, repeated for $E_{T\ rain}$ epochs. Meta-testing includes 600 tasks or randomly sampled episodes per dataset, with average accuracy reported for Derm7pt [6] and BloodMNIST [1] dataset.

5. RESULTS AND ANALYSIS

5.1 Comparison with SOTA Methods

We performed a comparative evaluation of proposed model's per-formance against multiple state-of-the-art (SOTA) methods.

5.1.1 Derm7pt. Figure 4 presents a comparative analysis of pro-posed method against various state-of-the-art approaches on the Derm7pt dataset. The bar charts depict 1-shot and 5-shot accu-racies across multiple methods, where each bar corresponds to a reported SOTA model, and the error bars represent performance variance. As shown in the 1-shot chart, most methods cluster be-tween 56–62%, with WRN-28-10 and DenseNet121 variants reach-ing around 61%, and Conv6 approaches plateauing below 63%. In contrast, proposed method's Conv4-based model (red bar) clearly rises above the group, achieving 67.18%, which is an approximate gain of 5% over the strongest WRN-28-10 baseline and more than 6% over Conv6. The 5-shot chart illustrates an overall upward shift in performance, with WRN-28-10 methods peaking at 79.83% and DenseNet121 around 79.18%, while Conv6 approaches remain be-low 77%. Proposed method's model maintains strong competitive performance at 78.20%, comparable to the heavy WRN-28-10 and DenseNet121 backbones.

Despite employing a lightweight Conv4 backbone with only 50,688 parameters, the proposed method consistently outperforms much larger models in the more challenging 1-shot setting while sustain-ing competitive results in the 5-shot scenario. Moreover, as high-lighted in the Baseline [19] work, such compact architectures are advantageous for feature extraction offering faster inference, lower memory requirements, and improved generalization making them particularly well-suited for resource-constrained clinical deploy-ment.

5.1.2 BloodMNIST. Figure 5 presents a comparative analysis of the proposed method against various state-of-the-art approaches on

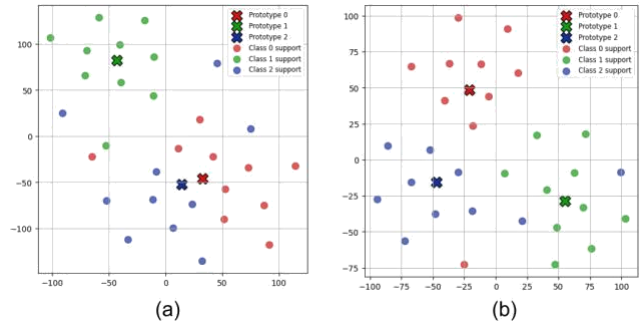


Fig. 6: t-SNE Visualization of (a) ProtoNet vs (b) Proposed approach on 3-way 10-shot classification in BloodMNIST dataset

the BloodMNIST dataset. The bar charts show 1-shot, 5-shot, and 10-shot accuracies, with each bar representing a reported SOTA model and the error bars indicating variance. In the 1-shot set-ting, most approaches fall between 55–63%, with ResNet-50 meth-ods performing strongly—Multi-ProtoNets achieves 63.10% and GDSL with ResNet-25 reaches 63.47%. The proposed Conv4-based model (red bar) delivers 63.39%, nearly matching the best heavy ResNet-50 baselines despite using only 50k parameters. In the 5-shot chart, overall performance rises, with LDP-Net (ResNet-
50) achieving 73.31% and Latent Augmentation reaching 70.22%. Yet, the proposed Conv4-based method clearly surpasses all, at-taining 77.17% and outperforming both ResNet-50 and ResNet-25 models. The 10-shot results highlight a similar trend: ResNet-50 methods plateau around 79–80% (e.g., Multi-ProtoNets at 80.01%), while some ResNet-25 approaches such as R2D2 reach
82.66%. In contrast, the proposed Conv4 model continues to im-prove and achieves the best overall accuracy of 83.06%, exceeding both heavy and mid-weight backbones.
Overall, while deep models like ResNet-50 (23.9M parameters) and ResNet-25 (15–17M parameters) perform competitively, their gains diminish at higher shots. Lightweight Conv4 methods like GDSL suffer sharp drops in accuracy (53.41% at 1-shot and only 63.81% at 10-shot). The proposed Conv4-based approach not only closes this gap but also establishes new performance highs, demonstrat-ing that extremely compact models can exploit scarce support ex-amples more effectively than heavier architectures, making them highly practical for resource-constrained clinical deployment.

5.2 t-SNE Visualization

To qualitatively assess the discriminative power of learned proto-types, we visualize the embedded representations of support and query samples for a representative 3-way 10-shot task on Blood-MNIST using t-SNE. As shown in Figure 6 (a) conventional Pro-totypical Network produces clusters that are both dispersed and partially overlapping indicating prototype ambiguity and weaker inter-class separation. In contrast, 6 (b) Selective Feature Repre-sentation in Prototypical Networks approach yields markedly more compact and well-separated clusters with each class forming a dis-tinct, tightly grouped manifold. This clear improvement stems from proposed method's channel-wise filtering, which suppresses noisy or redundant activations and retains only the most informative fea-ture maps for prototype construction. Consequently, the result-ing embeddings exhibit sharper decision boundaries and reduced intra-class variance, providing compelling gains in few-shot accu-racy achieved by proposed method.

6. CONCLUSION AND FUTURE DIRECTIONS

In this work, we introduced a lightweight yet effective refinement to prototypes of prototypical few-shot learning by Selective Fea-ture Representation, which explicitly filters support embeddings by selecting only the top-M feature maps per class using an ensem-ble of Global Average Pooling, Max Pooling, and Variance crite-ria. Built upon a compact CONV4 backbone, proposed approach achieves state-of-the-art performance on two challenging medical imaging benchmarks, yielding 67.18% (1-shot) on Derm7pt and up to 77.17% (5-shot) and 83.06% (10-shot) on BloodMNIST, outper-forming much larger architectures.
Looking ahead in future, although proposed selective feature re-finement strategy shows strong potential, there are several direc-tions to further improve and broaden its impact. First, to select more varied and discriminative feature maps, future work will fo-cus on employing additional statistical measures or alternative tech-niques alongside the current ensemble of Global Average Pooling, Max Pooling, and Variance. This would allow the identification of the most descriptive and representative feature maps that not only form stronger class prototypes but also highlight the most salient regions of medical images, thereby improving interpretability and classification accuracy. Second, we plan to extend this framework to a wider range of medical imaging datasets, particularly those that face resource constraints, limited annotations, and severe class im-balance. Validating proposed method's under such challenging con-ditions will help demonstrate its robustness and practical usefulness in real-world clinical scenarios where high-quality labeled data is scarce. Together, these directions aim to make selective feature re-finement a more general, adaptive, and clinically valuable solution for few-shot medical image analysis. Overall, selective feature re-finement represents a promising direction for enhancing prototype quality in few-shot medical imaging while maintaining efficiency and scalability.

7. REFERENCES

[1] Andrea Acevedo, Anna Merino, Santiago Alferez,´ Angel Molina, Laura Boldu,´ and Jose´ Rodellar. A dataset of micro-scopic peripheral blood cell images for development of auto-matic recognition systems. Data in Brief, 30:105474, 2020.

[2] Arijit Bera, Subhransu Kumar, and Ashwin Machanavajjhala. Exploring medical image mining using deep convolutional networks. In SIAM International Conference on Data Mining, pages 543–551, 2018.

[3] Runyang Cao, Yu Zhang, Song Bai, and Philip Torr. Comet: Concept-based prototypical networks for few-shot learning. In International Joint Conference on Artificial Intelligence, pages 2108–2114, 2021.

[4] Yujia Chen, Zhenguo Li, and Ruijia Wang. Feature-level meta-learning with attention for few-shot classification. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 36, pages 512–520, 2022.

[5] Andre Esteva, Brett Kuprel, Roberto A. Novoa, Justin Ko, Susan M. Swetter, Helen M. Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. In Nature, pages 115–118, 2017.

[6] J. Kawahara et al. Seven-point checklist and skin lesion clas-sification. IEEE JBHI, 2019.

[7] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks.

In International Conference on Machine Learning, pages 1126–1135, 2017.

[8] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, Franc¸ois Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. In International Conference on Machine Learning, pages 1180–1189, 2016.

[9] Xia He, Xue Yuan, Kai Zhang, Hu Li, and Wei Liu. Robust re-weighting prototypical networks for few-shot classification. In Proceedings of the 6th International Conference on Robotics and Artificial Intelligence (ICRAI '20), pages 125–132, Singapore, 2020. ACM.

[10] Timothy M. Hospedales, Antreas Antoniou, Paul Mica-helis, and Amos Storkey. Meta-learning in neural net-works: A survey. In International Conference on Learning Representations, 2020.

[11] Jieneng Hu and et al. Feature disentanglement and filter-ing for few-shot learning. In Proceedings of the International Conference on Learning Representations (ICLR), 2023.

[12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems, pages 1097–1105, 2012.

[13] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. In Proceedings of the 28th International Conference on Neural Information Processing Systems, pages 1–9, 2015.

[14] Qi Li, Hao Zhang, and Wei Wang. Learning to select informative features for few-shot learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 11835–11844, 2023.

[15] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud A.A. Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen A.W. M. van der Laak, Bram van Ginneken, and Clara I. Sanchez´. A survey on deep learning in medical image anal-ysis. In Medical Image Computing and Computer-Assisted Intervention – MICCAI, pages 295–307, 2017.

[16] Xiaolong Liu, Yifan Wang, and Lei Zhang. Weighted prototypical networks for few-shot learning. In AAAI Conference on Artificial Intelligence, pages 9319–9326, 2019.

[17] Yun Liu, Geert Litjens, and Bram van Ginneken. Deep learn-ing for content-based medical image retrieval: A performance study. In 2016 IEEE International Conference on Image Processing (ICIP), pages 1275–1279, 2016.

[18] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I. Jordan. Learning transferable features with deep adaptation networks. In International Conference on Machine Learning, pages 97–105, 2015.

[19] Xu Luo, Hao Wu, Ji Zhang, Lianli Gao, Jing Xu, and Jingkuan Song. A closer look at few-shot classification again. In International Conference on Machine Learning, pages 23103–23123. PMLR, 2023.

[20] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In ICLR Workshop, 2017.

[21] Jun Shu, Qi Xie, Lixuan Yi, Qian Zhao, Sanping Zhou, Zongben Xu, and Deyu Meng. Meta-weight-net: Learning an explicit mapping for sample weighting. In Advances in Neural Information Processing Systems, NeurIPS 2019, pages 1917–1928, 2019.

[22] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypi-cal networks for few-shot learning. In Advances in Neural Information Processing Systems, pages 4077–4087, 2017.

[23] Koichi Suzuki, Katsuyuki Mori, and Kunio Doi. Medical im-age sorting and retrieval by combining shape and texture fea-tures. In SPIE Medical Imaging, pages 978–985, 2016.

[24] Oriol Vinyals, Charles Blundell, Tim Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one-shot learning. In Advances in Neural Information Processing Systems, pages 3630–3638, 2016.

[25] Yongqiang Wang, Yikang Li, Kai Chen, Shiji Song, and Ying Huang. A survey on few-shot learning. In CVPR Workshops, pages 2866–2875, 2020.

[26] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In Advances in Neural Information Processing Systems, pages 3320–3328, 2014.

[27] Zhongzheng Yue, Qianru Sun, and Qi Yu. Compact fea-ture learning for prototypical few-shot classification. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), pages 10454–10463, 2023.