Towards Cross-Granularity Few-Shot Learning: Coarse-to-Fine Pseudo-Labeling with Visual-Semantic Meta-Embedding

Jinhai Yang, Hua Yang*, Lin Chen
Institute of Image Communication and Network Engineering, Shanghai Jiao Tong University, China
Shanghai Key Lab of Digital Media Processing and Transmission, Shanghai, China
MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University, China

Figure 1: Illustrating the cross-granularity few-shot image classification problem (along with a 3-way 1-shot testing case). (a) Training with only coarse class labels (in blue). (b) Testing on novel few-shot tasks with unseen fine class labels (in red).

ABSTRACT
Few-shot learning aims at rapidly adapting to novel categories with only a handful of samples at test time, which has been predominantly tackled with the idea of meta-learning. However, meta-learning approaches essentially learn across a variety of few-shot tasks and thus still require large-scale training data with fine-grained supervision to derive a generalized model, thereby involving prohibitive annotation cost. In this paper, we advance the few-shot classification paradigm towards a more challenging scenario, i.e., cross-granularity few-shot classification, where the model observes only coarse labels during training while is expected to perform fine-grained classification during testing. This task largely relieves the annotation cost since fine-grained labeling usually requires strong domain-specific expertise. To bridge the cross-granularity gap, we approximate the fine-grained data distribution by greedy clustering of each coarse-class into pseudo-fine-classes according to the similarity of image embeddings. We then propose a meta-embedder that jointly optimizes the visual- and semantic-discrimination, in both instance-wise and coarse class-wise, to obtain a good feature space for this coarse-to-fine pseudo-labeling process. Extensive experiments and ablation studies are conducted to demonstrate the effectiveness and robustness of our approach on three representative datasets.

CCS CONCEPTS
• Computing methodologies → Learning paradigms; Image representations; Transfer learning; • Information systems → Similarity measures.

KEYWORDS
meta-learning, few-shot learning, cross-granularity learning, inexact supervision, visual-semantic embedding

1 INTRODUCTION
As a hallmark of intelligence, humans can easily learn new concepts with scarce samples. In stark contrast, recent advances in deep learning models usually demand immense quantities of data with...
To tackle this new problem, we propose a Visual-Semantic Meta-Embedder (VSME) with a Coarse-to-Fine (C2F) pseudo-labeling algorithm to generate the approximate fine-grained training classes. Unlike the pseudo-labeling technique widely used in the field of semi-supervised learning [24, 31, 53, 65], our C2F pseudo-labeling produces fine labels for coarse-labeled data. We cluster each coarse class into pseudo-fine-classes via greedy similarity matching to establish a task distribution of analogous granularity to the target testing tasks, as shown in Fig. 2 (d). Recent works have shown that the fast adaptation of ML crucially depends on feature reuse [44], and a good feature embedding trained on the merged base-classes performs favorably in the standard ML setting [5, 8, 60, 63]. Therefore, we present the VSME to learn from all training samples and then generate image embeddings for the similarity matching, along with a Base-Embedder as the downstream few-shot learner (i.e., both the meta- and base-learner of the FSL stage in Fig. 2 (d)).

We jointly optimize the visual- and semantic-discrimination between the VSME features, as illustrated in Fig. 3. Instance-based embedding methods [11, 64] take each instance as an individual class. [66] utilizes visual transformations to augment the samples and then conducts contrastive learning within mini-batches. It focuses on the visual side and helps exploit the internal structure of each coarse class. Supervised embedding [18, 39, 40, 60] explicitly maximizes inter-class variation while minimizes intra-class variation. These methods provide more semantic information and help explore the external relation between coarse classes, but also make the fine-grained classes inside one coarse class inseparable unfavorably. In this paper, we combine the complementary strengths of instance-wise supervision and coarse-class labels to integrate both visual- and semantic-discrimination.

Our contributions are summarized as follows: (1) We propose the cross-granularity few-shot classification task and cast it as the inexact-supervised meta-learning problem, which learns from coarse labels for few-shot testing on unseen fine classes. (2) We devise a Visual-Semantic Meta-Embedder to encode images and the Coarse-to-Fine pseudo-labeling that performs similarity matching with the embeddings to generate pseudo-fine-classes. (3) We conduct extensive experiments and analysis to show the effectiveness and robustness of the proposed approach on three datasets.
create proxy tasks [9, 43, 68] to drive feature learning without any external guidance, while instance-wise contrastive methods treat each sample as a different class [11, 64], and are usually built on the intuition that different augmentations of an image should be recognized as the same instance [4, 19, 66]. As shown in Fig. 3, we combine the strength of supervised (with coarse label) and instance-based methods to work in synergy.

**Meta-learning with limited supervision.** Although aiming at few-shot testing, most ML methods still heavily depend on large-scale fine-grained annotations. Recently, some studies have begun to explore meta-learning with different types of limited annotations. [47] pioneered semi-supervised meta-learning, which mixes a small labeled support set with a large pool of unlabeled data, and proposed an extension of Prototypical Networks [55] to better utilize the unlabeled set. Under the same setting, [35] introduced a self-training strategy and a cherry-picking network to select and annotate the unlabeled set. Unsupervised meta-learning purely learns from unsupervised data, but, meanwhile suffers dramatic performance drops due to the lack of annotations. Existing works attempted to synthesize training tasks by clustering embeddings [22] or random sampling and augmentation [26]. In our framework, we use only coarse labels to provide moderate supervision to balance the trade-off between annotation overhead and performance.

**Learning with inexact supervision.** Since the supervision information given by coarse labels is not as exact as desired, it is well-known as inexact supervision, one of the most common types of weak supervision [70]. Some works [17, 48] have considered learning problems entailing the mixture of coarse and fine labels, training the model with either balanced [58] or unbalanced [49] batches containing both coarse- and fine-labeled data. In the context of FSL, there are few works concentrated on coarse-grained information. [36] proposed a prototype propagation mechanism to effectively exploit the hierarchical labels of ImageNet [7] and learn a multi-level directed acyclic prototype graph. However, this method treats coarse labels as auxiliary information for few-shot classification tasks and assumes that the class hierarchy is known in advance (even during testing). Differently, we observe only coarse labels of base classes and the target classes belong to unseen coarse classes. To our best knowledge, we are the first study to leverage pure inexact supervision in FSL.

## 3 PROBLEM FORMULATION

In standard few-shot classification, given a meta-training set \(\mathcal{D}_{train}\) containing samples of base classes \(c_{base}^{fine}\), the goal is to achieve high accuracy on classification episodes sampled from a testing set \(\mathcal{D}_{test}\) that consists of novel classes \(c_{novel}^{fine}\). The novel classes are totally different from the base classes, i.e., \(c_{base}^{fine} \cap c_{novel}^{fine} = \emptyset\). An \(N\)-way \(K\)-shot episode is defined as a tuple \((S_{support}, S_{query})\), where the support set \(S_{support}\) contains \(N\) classes with \(K\) samples in each class and the query set \(S_{query}\) contains the same \(N\) classes with \(Q\) unlabeled samples in each class. The episodic objective is to correctly classify the \(N \times Q\) query samples into the \(N\) classes, taking \(S_{support}\) as a reference.

In cross-granularity few-shot classification, the training set \(\mathcal{D}_{train} = \{(x_i, y_i)\}\) includes pairs of images \(x_i \in X\) and coarse
labels \( \tilde{y}_i \in C^{\text{coarse}}_\text{base} \) rather than the fine label \( y_i \in C^{\text{fine}}_\text{base} \). Taking ImageNet [7] as an example, instead of being labeled exactly, the images of class “lamp” and “bookcase” may share the coarse-category label “furnishing”. Besides the fine labels, the coarse labels between base classes and novel classes should also be non-overlapping, that is, \( C^{\text{base}} \cap C^{\text{coarse}} \neq \emptyset \), making CGFSC more challenging than standard FSL. Notably, in CGFSC the goal is still to minimize the few-shot test error on the fine novel classes \( C^{\text{fine}}_\text{novel} \).

### 4 METHOD

#### Framework overview
An overview of the proposed framework is illustrated in Fig. 4. The training process consists of three procedures: (1) **Meta-Embedder Training**: Train a Convolutional Neural Network (CNN) as the meta-embedding function \( f_\phi \) with joint instance-wise supervision and coarse class-wise supervision on the meta-training set \( D^{\text{train}} \). (2) **C2F Pseudo-Labeling**: Use \( f_\phi \) to derive image embeddings for all samples in \( D^{\text{train}} \) and then conduct similarity matching to pseudo-label each coarse class \( C^{\text{coarse}}_\text{base} \) into pseudo-fine-classes \( C^{\text{pseudo}}_\text{base} \), according to the similarity of the corresponding features. (3) **Base-Embedder Training**: Train another CNN as the base-embedding function \( f_\psi \) on the pseudo-dataset \( D^{\text{pseudo}}_\text{train} \) to excavate transferable knowledge. Last, for the few-shot testing, we take the fixed \( f_\phi \) as a feature extractor and utilize a nearest-prototype classifier with cosine similarity to predict the fine label for \( S_{\text{query}} \) using a fine-labeled \( S_{\text{support}} \).

#### 4.1 Visual-Semantic Meta-Embedder

In this section, we elaborate on the details of the proposed VSME, which approximates both visual discrimination and semantic discrimination in the latent space. As shown in Fig. 5, VSME simultaneously exploits two types of supervision: (1) instance-wise supervision and (2) coarse class-wise supervision. Although they share the same objective of pulling similar images together and pushing dissimilar ones away in a compact space, they encourage feature discrimination on quite different aspects. The instance-wise supervision imposes consistency regularization [1] to ensure the model produces identical output distribution when its inputs are visually perturbed, hence it focuses more on the visual side. Differently, the coarse class-wise supervision enforces the feature vectors discriminative in the semantic aspect.

We use four methods (RandomResizedCrop, ColorJitter, RandomGrayscale, RandomHorizontalFlip) to augment the images. For a coarsely-labeled sample \( (x, \tilde{y}) \in D^{\text{train}} \), let \( \tilde{x} \) denote the augmented sample. For simplicity, let \( f \) and \( f \) denote their embedding features. As shown in Fig. 5, an image \( x \) is first encoded by \( f_\phi \), next passed through the global average pooling (GAP) to take the average of each feature map, and then converted to a low-dimensional embedding \( f \in \mathbb{R}^M \) with a fully-connected layer FC1 (assuming the output dimension is \( M \)). The embeddings are \( \ell_2 \) normalized, thus \( \| f \|_2 = 1 \), therefore the cosine similarity can be efficiently calculated by the dot product. Following [64, 66], we use the softmax function with cosine similarity on top of the embeddings to match the instances. For the \( i \)-th sample \( x_i \) in a batch \( B \), the probability of classifying \( \hat{x}_i \) as the original instance \( x_i \) is defined by

\[
P(x_i | \hat{x}_i) = \frac{\exp \left( \frac{\langle f_\phi^T f_\phi \rangle}{\tau} \right)}{\sum_{x_j \in B} \exp \left( \frac{\langle f_\phi^T f_\phi \rangle}{\tau} \right)} ,
\]

where \( \tau \) is the temperature [15, 16] scaling the entropy of the output probability distribution. The probability of another instance \( x_j \) being falsely recognized as the \( i \)-th instance is

\[
P(x_i | x_j) = \frac{\exp \left( \frac{\langle f_\phi^T f_\phi \rangle}{\tau} \right)}{\sum_{x_k \in B} \exp \left( \frac{\langle f_\phi^T f_\phi \rangle}{\tau} \right), j \neq i}.
\]

The **visual-side** negative log-likelihood loss within the batch is

\[
L_{\text{vis}} = - \sum_{x_i \in B} \log P(x_i | \hat{x}_i) - \sum_{x_i \in B} \sum_{x_j \in B, x_j \neq x_i} \log (1 - P(x_i | x_j)) .
\]

To harness the coarse labels \( \tilde{y} \in C^{\text{coarse}}_\text{base} = \{ 1, 2, \ldots, C \} \) for the semantic discrimination property, the embedding features \( f \) are sent to a linear classifier (FC2) \( W = [W_1, W_2, \ldots, W_C] \in \mathbb{R}^{M \times C} \) with a softmax layer for classification of the coarse classes. The probability of \( x_i \) belonging to class \( c \) is defined by

\[
P(c | x_i) = \frac{\exp \left( W_c^T f_\phi \right)}{\sum_{k=1}^C \exp \left( W_k^T f_\phi \right)} .
\]
On the other hand, the augmented image $\hat{x}_i$ should be classified into the same class as that of the original one. The probability of the augmented image $\hat{x}_i$ being classified to class $c$ is denoted by

$$P(c|\hat{x}_i) = \frac{\exp \left( W^T_{\gamma} \hat{x}_i \right)}{\sum_{k=1}^{C} \exp \left( W^T_{\gamma} \hat{x}_i \right)}. \quad (5)$$

Let $y_{i,j}$ denote the $j$-th class label of instance $x_i$. The semantic-side negative log-likelihood loss within the batch is

$$L_{\text{sem}} = - \sum_{x_i \in \mathcal{B}} \sum_{y_{i,j}} \log [P(j|x_i) P(j|\hat{x}_i)]. \quad (6)$$

The joint loss of the VSME is the sum of the two loss items,

$$L_{\text{meta}}(\mathcal{B}; \theta) = L_{\text{vis}} + L_{\text{sem}}. \quad (7)$$

and the optimal parameters are $\hat{\theta}^* = \arg \min_{\theta} \mathbb{E}_{\mathcal{B}} \left[ L_{\text{meta}}(\mathcal{B}; \theta) \right].$

### 4.2 Coarse-To-Fine Pseudo-Labeling

Unlike the pseudo-labeling in the usual sense [31, 35], we propose the coarse-to-fine pseudo-labeling that takes as input an image with a coarse label $(x, y) \in \mathcal{D}_{\text{train}}$ and assigns a fine-grained pseudo-label to it. As outlined in Alg. 1, this process continually selects random images as the root samples and then makes locally-optimal choices to produce the pseudo-sub-categories, until the pseudo-labeled training dataset $\mathcal{D}_{\text{train}}^{\text{pseudo}}$ is entirely derived. In this process, we still conduct image matching with cosine similarity (dot product) on top of the corresponding features encoded by the fixed $f_{\phi^*}$.

### 4.3 Base-Embedder

Once all coarse classes in $\mathcal{D}_{\text{train}}$ are divided into fine-grained sub-categories via C2F pseudo-labeling, we train another embedding model $f_{\theta}$ as the few-shot learner with the pseudo-dataset $\mathcal{D}_{\text{train}}^{\text{pseudo}}$. As shown in Fig. 4, the training of $f_{\theta}$ takes place in two consecutive stages: (1) pre-training with all pseudo-fine-classes and (2) episodic training on few-way few-shot episodes sampled from $\mathcal{D}_{\text{train}}^{\text{pseudo}}$.

---

**Algorithm 1 Coarse-To-Fine (C2F) Pseudo-Labeling**

**Require:** $N_c$: Number of samples per pseudo-fine-class  
**Require:** $C$: Number of coarse classes, $C = \frac{C_{\text{base}}}{C_{\text{coarse}}}$  
**Require:** $\mathcal{D}_{\text{train}} = \{(x_i, y_i)\}$: Training data with coarse labels  
1: Initialize the pseudo-dataset $\mathcal{D}_{\text{train}}^{\text{pseudo}} = \emptyset$
2: for class $c$ from 1 to $C$
3: Retrieval $\hat{D}_c = \{x_j | y_j = c, (x_j, y_j) \in \mathcal{D}_{\text{train}}\}$, let $N = |\hat{D}_c|$  
4: Get the embeddings $F_c \in \mathbb{R}^{M \times N}$ of the samples in $\hat{D}_c$
5: Obtain the similarity matrix $S = F_c^T F_c \in \mathbb{R}^{N \times N}$
6: while number of the remaining samples of $\hat{D}_c$ $\geq N_c$
7: Randomly select $x_j$ from $\hat{D}_c$ as the root of new class $c_n$
8: Retrieval $(x_k)_{k=1}^{N_{c_n}}$ with top similarity to $x_j$
9: $\mathcal{D}_{\text{train}}^{\text{pseudo}} \leftarrow \mathcal{D}_{\text{train}}^{\text{pseudo}} \cup \{(x_j, c_n)\}$  
10: Remove the selected $N_c$ samples from $\hat{D}_c$
11: end while
12: Discard the rest of the samples of $\hat{D}_c$
13: end for

In the pre-training stage, our goal is to learn a set of parameters $\theta$ as the initialization of $f_{\theta}$ via a classification task on the complete pseudo-dataset. Let $L_{ce}$ denotes the cross-entropy loss, the optimization objective is given by

$$\theta^* = \arg \min_{\theta} L_{ce} \left( \mathcal{D}_{\text{train}}^{\text{pseudo}}, \theta \right). \quad (8)$$

In the episodic-training stage, our goal is to enhance the fast adaptability of $f_{\theta}$ initialized by $\theta^*$ via training across a variety of few-shot episodes. In each episode, a subset of $\mathcal{D}_{\text{train}}^{\text{pseudo}}$ is sampled to construct a few-shot task $T = (S_{\text{query}}^{\text{pseudo}}, S_{\text{support}}^{\text{pseudo}})$. The embedding model $f_{\theta}$ makes predictions for the unlabeled query set using the support set (see Sec. 4.4). Formally, the training objective is

$$\psi^* = \arg \min_{\psi} \mathbb{E}_{T} \left[ L_{ce} \left( S_{\text{query}}^{\text{pseudo}}, \psi, S_{\text{support}}^{\text{pseudo}} \right) \right]. \quad (9)$$
4.4 Few-shot Testing

Different from prior work that fine-tunes the embedding model [8] or trains a new classifier (e.g., a Fully-Connected layer [5], a Logistic Regression model [60] or a Support Vector Machine (SVM) [32]) as the base-learner for each task, we keep the \( f_p \) unchanged and adopt a non-parametric nearest-prototype classifier for few-shot testing. Given an \( N \)-way \( K \)-shot task, let \( S_c \subset S_{\text{support}} \) denote the samples in class \( c \), the class prototype \( v_c \) is computed by

\[
v_c = \frac{1}{K} \sum_{x_i \in S_c} f_p(x_i).
\]

We then predict the probability that a query sample \( x \) belongs to class \( c \) using the softmax function on top of the cosine similarity (with temperature \( \tau \)) between the embedding vector of \( x \) and the class prototype \( v_c \). Formally,

\[
P(c|x) = \frac{\exp \left( v_c^T f_p(x)/\tau \right)}{\sum_{k=1}^{N} \exp \left( v_k^T f_p(x)/\tau \right)}.
\]

5 EXPERIMENTS

5.1 Datasets

We conduct experiments on three popular benchmark FSL datasets: tieredImageNet [47], FC100 [41] and Omniglot [30]. These datasets are all hierarchical since images are annotated by not only fine labels but also higher-level coarse labels, thus they are suitable as a benchmark for the CGFSC task. To simulate the situation of inexact supervision, we observe only coarse labels during training. tieredImageNet is a subset of ILSVRC-12 [50], containing 608 classes from 34 super-categories according to the ImageNet [7] hierarchy. The super-categories are divided into 20/6/8 splits for training, validation, and testing respectively, ensuring the training classes to be distinct enough from the testing classes.

FC100 is a derivative of CIFAR-100 [28] with the classes organized in a similar way to tieredImageNet. Omniglot contains handwritten characters from various alphabets (like Latin and Korean). Likewise, we modify the Omniglot by partitioning different alphabets into different splits. Concretely, we use all alphabets in the background split for training, seven alphabets (Manipur, Atemayar, Qelisayer, Sytheti, Kebie, Gurmukhi, ULOG, Old_Church_Slavonic_(Cyrillic)) for validation, and the other alphabets are left for testing. Detailed statistics of the three datasets are summarized in Tab. 1.

Table 1: Dataset Statistics. Class numbers are given according to training/validation/testing splits. Coarse classes in different splits are non-overlapping. Our method largely relieves the annotation burden, as the number of coarse classes is usually far smaller than that of fine classes.

| Dataset                      | tieredImageNet | FC100 | Omniglot |
|------------------------------|----------------|-------|----------|
| Coarse classes               | 20/6/8         | 12/4/4| 30/7/13  |
| Fine classes                 | 351/97/160     | 60/20/20 | 964/236/423 |
| Images per (fine) class      | ~1280          | 600   | 20       |
| Resolution                   | 84x84          | 32x32 | 28x28    |

5.2 Implementation Details

Network architecture. (1) Regarding the meta-embedder. We use a standard ResNet-18 [20] as the backbone, following previous works in unsupervised embedding learning [64, 66]. (2) Regarding the base-embedder. For Omniglot, we apply ConvNet-4 as the few-shot learner following [13, 33, 62, 67]. ConvNet-4 is composed of 4 modules, each with a 64-filter 3 × 3 convolutional layer followed by batch normalization [23] and a ReLU activation. A 2 × 2 max-pooling layer is applied after each module. For tieredImageNet and FC100, we adopt ResNet-12 as commonly-used in [32, 37, 41, 46, 60]. ResNet-12 contains 4 residual blocks, each having 3 modules, and the first 3 blocks are followed with a 2×2 max-pooling. The number of filters in the blocks is (64,160,320,640). (3) For all models, the feature maps produced by the backbone are aggregated by a GAP layer to derive low-dimensional embeddings.

Training details. We implement our model using PyTorch [42]. We set the embedding dimension \( M \) to 128 and the temperature \( \tau \) to 0.1. The number of images per pseudo-fine-class \( N_f \) for tieredImageNet is 1500, for FC100 is 600, and for Omniglot is 20. The optimizer is SGD with momentum 0.9, weight decay \( 5e^{-4} \), and learning rate decay factor 0.1. (1) Regarding the meta-embedder. All experiments are trained for a total of 200 epochs, and the learning rate starts from 0.3 and decays at epoch 120 and 160. A mini-batch contains 128 samples. (2) Regarding the base-embedder. In the pre-training stage, we train 60 epochs for tieredImageNet, 90 epochs for FC100 and Omniglot. The learning rate is initialized as 0.05 and decayed twice after the first 30 epochs. The batch size is 64. In the episodic training stage, we fix the learning rate to 0.001 and train all models on 2000 pseudo-labeled episodes.

Evaluation metric. To assess the cross-granularity few-shot performance, we evaluate our method on 1000 fine-labeled episodes and report the mean accuracy with 95% confidence intervals. Notably, whereas our models learn from only coarse labels, this evaluation protocol is the same as the setting of standard FSL.
Table 3: Comparison to fine-to-fine state-of-the-arts and other coarse-to-fine baselines on FC100. Average 5-way accuracy (%) with 95% confidence intervals. †Upper bound of the coarse-to-fine experiments. All included methods are with ResNet-12 as the backbone of the few-shot learner.

| Method                                | 1-shot  | 5-shot  |
|----------------------------------------|---------|---------|
| Prototypical Nets [55]                 | 37.50 ± 0.60 | 52.50 ± 0.60 |
| TADAM [41]                             | 40.10 ± 0.26 | 56.10 ± 0.40 |
| MetaOptNet [32]                        | 41.10 ± 0.60 | 55.50 ± 0.60 |
| RFS-Simple [60]                        | 42.60 ± 0.70 | 59.10 ± 0.60 |
| Our Baseline †                         | 42.39 ± 0.64 | 58.91 ± 0.65 |

Table 4: Comparison to fine-to-fine state-of-the-arts and other coarse-to-fine baselines on Omniglot. Average 5-way accuracy (%) with 95% confidence intervals. †Upper bound of the coarse-to-fine experiments. All included methods are with ConvNet-4 as the backbone of the few-shot learner.

| Method                                | 1-shot  | 5-shot  |
|----------------------------------------|---------|---------|
| Baseline w/ Coarse Labels              | 37.43 ± 0.66 | 52.79 ± 0.21 |
| Baseline + C2F w/ Pixels               | 33.99 ± 0.19 | 45.31 ± 0.43 |
| VSME w/o C2F (Ours)                    | 38.22 ± 0.61 | 53.54 ± 0.60 |
| Baseline + C2F w/ VSME (Ours)          | 39.57 ± 0.23 | 54.63 ± 0.63 |

5.3 Results on Standard Benchmarks

Baseline models. Since the cross-granularity few-shot classification is a new task, we construct a diverse set of baselines to examine the effectiveness of the proposed method. First, we present several state-of-the-art FSL results for each dataset (in the fine-to-fine setting) as a reference. Second, we also evaluate our few-shot learner (the Base Embedder) under the F2F setting as the upper bound experiments of the C2F counterparts. Last, we considered the following alternatives as the baselines under the coarse-to-fine setting: (1) Baseline w/ Coarse Labels. Train BE with coarse labels directly (without C2F pseudo-labeling). (2) Baseline + C2F w/ Pixels. Conduct C2F pseudo-labeling with pixel-level features rather than embeddings produced by VSME. (3) VSME w/o C2F. Directly evaluate the VSME with the nearest-prototype classifier rather than using C2F with VSME to generate the pseudo-fine-classes for the downstream few-shot learner to learn from. (4) Baseline + C2F w/ VSME. Our full approach that conducts C2F pseudo-labeling with VSME embeddings and trains a BE as the few-shot learner. For fairness, we ensure that all experiments on each dataset are carried out with the same backbone architecture of the few-shot learner, and also keep the hyperparameters the same as possible.

Results. The results on tieredImageNet, FC100, and Omniglot are shown in Tab. 2, 3, 4 respectively. On all datasets, we observe that our approach consistently outperforms the C2F baselines with a large margin. Especially in the 1-shot setting, which is more challenging compared to the 5-shot setting, we significantly improve the accuracy by 2-16% across all datasets. Notably, our results are also comparable to the F2F state-of-the-arts. More specifically, our method even outperforms [55] with the same backbone on FC100 by 2.07% in the 1-shot setting whereas for the 5-shot setting it improves the score by 2.13%. In contrast, when C2F pseudo-labeling with pixels, the performance is worse than training with coarse labels, indicating the vital importance of VSME in our method. We also notice that VSME w/o C2F achieves consistent improvements over the other C2F baselines, preceded only by our full model.

5.4 Ablation Study

In this section, we investigate the contribution of each component. Effect of the episodic training. Episodic training was widely used in the FSL problems [33, 55, 62], but recently some works [60, 63] prefer to merge all episodes together into a single classification task for model training. In this work, we use it as a warm-up stage
to get the model acquainted with tackling the FSL problems after pre-training on the merged task. As shown in Fig. 6, in all experiments the episodic training stage consistently improves the 5-way classification accuracy, especially for the 1-shot cases.

**Effect of the components of VSME.** To show the importance of both the visual- and semantic-discrimination, we evaluate the performance by removing the visual-side item (instance-wise supervision) or the semantic-side item (coarse class-wise supervision) from the full model (Baseline + C2F w/ VSME). The results are shown in Tab. 5 with the 95% confidence interval omitted due to space limitation. We observe that abandoning either of the discrimination would decrease the few-shot classification accuracy on both datasets. In particular, visual discrimination contributes the most. We suspect that the reason is that the instance-wise supervision applied for visual discrimination is far stricter than the coarse class-wise supervision for semantic discrimination, as it takes each instance as a separate class. We also note that these weakened versions still outperform the “Baseline + C2F w/ Pixels” baselines in Tab. 2 and Tab. 4. Another interesting phenomenon is that although when only semantic discrimination is left, the final performance is inferior to "Baseline w/ Coarse Labels"; it still brings additional gains than applying visual discrimination only.

### 5.5 Further Analysis

**Effect of the few-shot learner.** Since we generate pseudo-fine-labeled datasets as the intermediate results, our framework can be integrated with other FSL algorithms. Here, we choose [13, 32, 55] to study the effect of the few-shot learner. We first conduct C2F pseudo-labeling on tieredImageNet to generate a pseudo-dataset and then train different few-shot learners on it. As reported in Tab. 6, the methods using shallower backbone (ConvNet-4) suffer less performance drop facing the cross-granularity transfer challenge, but meanwhile, show lower C2F accuracy. The performance of [32] dramatically drops because it reinitializes a maximum-margin classifier (SVM) for each task, easily misled by the inaccurate pseudo-labels.

**Robustness analysis.** As described in Alg. 1, the C2F pseudo-labeling algorithm randomly selects from the unlabeled images as the root to establish a new pseudo-fine-class, involving a series of random sampling. The number of samples per pseudo-fine-class \(N_s\) is another key parameter in the C2F algorithm. To verify the robustness of our method, we conduct experiments across 3 different

### ACKNOWLEDGMENTS

This work was funded by National Natural Science Foundation of China (NSFC, Grant No. 61771303), Science and Technology Commission of Shanghai Municipality (STCSM, Grant Nos. 19DZ1209303, 20DZ1200203, 2021SHZDZX0102), and SJTUYitu/Thinkforce Joint Laboratory for Visual Computing and Application.
