

# MetaFBP: Learning to Learn High-Order Predictor for Personalized Facial Beauty Prediction

Luojun Lin  
Fuzhou University  
Fuzhou, China  
linluojun2009@126.com

Zhifeng Shen  
Fuzhou University  
Fuzhou, China  
shen\_zhifeng@outlook.com

Jia-Li Yin  
Fuzhou University  
Fuzhou, China  
jlyin@fzu.edu.cn

Qipeng Liu  
Fuzhou University  
Fuzhou, China  
cometq@foxmail.com

Yuanlong Yu\*  
Fuzhou University  
Fuzhou, China  
yu.yuanlong@fzu.edu.cn

Weijie Chen\*  
Zhejiang University & Hikvision  
Research Institute  
Hangzhou, China  
chenweijie@zju.edu.cn

## ABSTRACT

Predicting individual aesthetic preferences holds significant practical applications and academic implications for human society. However, existing studies mainly focus on learning and predicting the commonality of facial attractiveness, with little attention given to *Personalized Facial Beauty Prediction* (PFBP). PFBP aims to develop a machine that can adapt to individual aesthetic preferences with only a few images rated by each user. In this paper, we formulate this task from a meta-learning perspective that each user corresponds to a meta-task. To address such PFBP task, we draw inspiration from the human aesthetic mechanism that visual aesthetics in society follows a Gaussian distribution, which motivates us to disentangle user preferences into a commonality and an individuality part. To this end, we propose a novel MetaFBP framework, in which we devise a universal feature extractor to capture the aesthetic commonality and then optimize to adapt the aesthetic individuality by shifting the decision boundary of the predictor via a meta-learning mechanism. Unlike conventional meta-learning methods that may struggle with slow adaptation or overfitting to tiny support sets, we propose a novel approach that optimizes a high-order predictor for fast adaptation. In order to validate the performance of the proposed method, we build several PFBP benchmarks by using existing facial beauty prediction datasets rated by numerous users. Extensive experiments on these benchmarks demonstrate the effectiveness of the proposed MetaFBP method.

## CCS CONCEPTS

• Information systems → Personalization; • Computing methodologies → Transfer learning.

\*Y. Yu and W. Chen are corresponding authors.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

MM '23, October 29–November 3, 2023, Ottawa, ON, Canada

© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.  
ACM ISBN 979-8-4007-0108-5/23/10...\$15.00  
<https://doi.org/10.1145/3581783.3612319>

## KEYWORDS

Meta Learning; Dynamic Network; Personalized Recommendation; Facial Beauty Prediction

### ACM Reference Format:

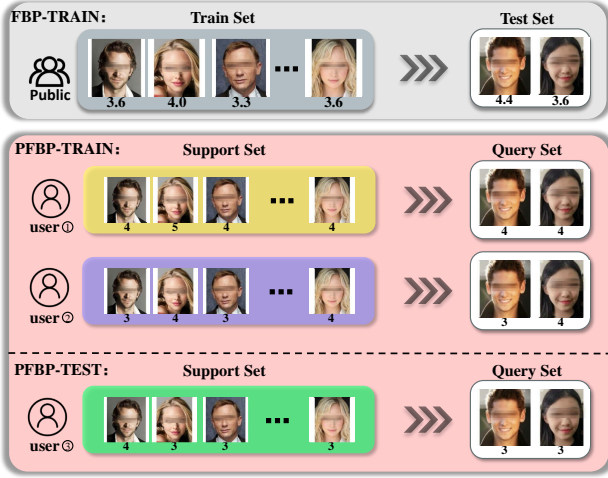
Luojun Lin, Zhifeng Shen, Jia-Li Yin, Qipeng Liu, Yuanlong Yu, and Weijie Chen. 2023. MetaFBP: Learning to Learn High-Order Predictor for Personalized Facial Beauty Prediction. In *Proceedings of the 31st ACM International Conference on Multimedia (MM '23)*, October 29–November 3, 2023, Ottawa, ON, Canada. ACM, New York, NY, USA, 9 pages. <https://doi.org/10.1145/3581783.3612319>

## 1 INTRODUCTION

Facial Beauty Prediction (FBP) has allured many research interests in recent years. Most studies aim to develop models that can accurately judge facial beauty in line with the average aesthetic preferences of a large population of users, using average ratings as the supervision signals for model learning [6, 27, 28]. However, these studies merely focus on the commonality and overlooks the highly subjective nature of human aesthetic perception, as illustrated in Figure 1, where each facial image is rated by different users with various attractiveness scores. Therefore, the subjective nature of facial aesthetics should be taken into account to develop more accurate predictions of facial attractiveness for different users.

To fill the current gap in this area, we delve into *Personalized Facial Beauty Prediction* (PFBP) in this paper. *The objective of this task is to make consistent aesthetic judgments with a specific user by previously requiring the user to rate a few facial images.* A highly-performing PFBP model has significant practical applications in various online systems, such as social recommendation systems or make-up recommendation systems [7]. As shown in Figure 1, the recommendation system requires each user to label a few facial images to adapt the PFBP model so that it can quickly capture the aesthetic preference and then send the top-ranked recommended faces from the image gallery to the target user. From this perspective, PFBP is expected to possess a fast adaptation ability for each user preference with limited labeled data.

Considering the user-adaptive and data-limited properties of the PFBP task, it intuitively motivates us to reformulate PFBP from a few-shot learning perspective [14]. Specifically, each individual user corresponds to a meta-task consisting of a support set and a query set. The training and evaluation of the PFBP model follow

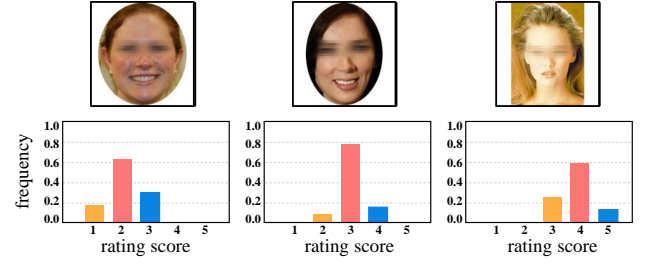


**Figure 1: The difference between FBP and PFBP. The conventional FBP only gives an average beauty score of the public for a facial image, but PFBP provides different beauty scores for each facial image according to user preference. Note that each face has been locally pixelated for privacy protection.**

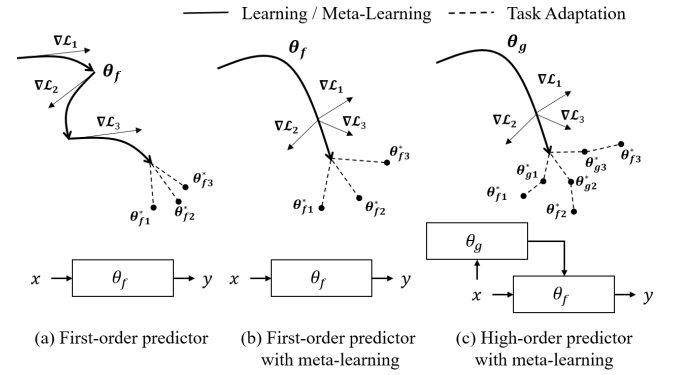
the meta-training and meta-testing stages in few-shot learning. Nevertheless, there are still two main differences between PFBP and conventional few-shot learning tasks:

- 1) In conventional few-shot learning tasks, the categories of each meta-task are different. The training goal is to quickly adapt the model from base to novel categories with limited training data. Unlike them, the categories of each meta-task in PFBP are fixed. The range of attractiveness score is shared among different meta-tasks. The training objective is not to adapt to novel categories but to novel users with specific aesthetic preferences.
- 2) In conventional few-shot learning tasks, the labels of images are fixed across meta-tasks, *e.g.*, the label of a cat image always belongs to “cat” and cannot be changed to other categories. However, in PFBP, the attractiveness score of a facial image will change across different meta-tasks because users have different aesthetic preferences and thus give different ratings to the same image.

The main challenge in PFBP is the subjective nature of user ratings, *i.e.*, the changeability of image labels across meta-tasks, which never occurs in previous image recognition tasks. It urges a rather strong adaptability of the PFBP model that can forget the image labels seen in previous meta-tasks and adapt to current meta-task quickly with limited labeled data. Despite the variability of ratings for facial images, we have observed that the attractiveness score of an image rated by a population of users tends to follow a Gaussian distribution, as shown in Figure 2. That is, the population aesthetic tends to be consistent while the personalized preference is fluctuated around the population aesthetic. This phenomenon can be attributed to the objective part of human aesthetic perception, namely aesthetic commonality, which plays a crucial role in working alongside the subjective part, aka aesthetic personality.



**Figure 2: The rating distributions of three randomly-selected images rated by a population of volunteers. The aesthetic preference roughly follows a Gaussian distribution.**



**Figure 3: The comparison of different learning paradigms. (a) Common learning paradigm. (b) Conventional meta-learning paradigm. (c) Our proposed paradigm that involves learning to learn high-order predictor for fast adaptation.**

Motivated by the observations, we propose to disentangle the personalized preference into a commonality part and a personality part from the network architecture perspective in this paper. The PFBP model is constructed with a universal feature extractor that represents aesthetic commonality and a personalized predictor that represents aesthetic personality. Specifically, the feature extractor is supervised by the average rated score which is similar to the training paradigm of a common FBP model, while the predictor is trained using individual rated score under a meta-learning paradigm. The predictor is expected a fast adaptation ability, but using conventional meta-learning paradigms are usually trapped in slow adaptation or over-fitting the tiny support set. To enhance its adaptation ability, we introduce learning-to-learn paradigm into a high-order predictor. Compared with the conventional predictor, aka first-order predictor, which is simply implemented by a fully-connected layer, the high-order predictor possesses a more powerful adaptation ability, by using a shallow parameter-generator to twist the weights of the predictor based on the input. Based on such architecture design, we further optimize the generator via a gradient-based meta-learning approach to form a meta-generator. Figure 3 illustrates the advantage of the proposed method, where the meta-generator can twist the weights of the high-order predictor quadratically for faster adaptation.

To stress the effectiveness of the proposed framework, termed *MetaFBP*, we establish several PFBP benchmarks based on the existing FBP datasets varied from small, medium to large scales, including PFBP-SCUT500, PFBP-SCUT5500, and PFBP-US10K. We conduct extensive experiments on these newly-established PFBP benchmarks, and the experimental results demonstrate that our method significantly outperforms the conventional meta-learning approaches. To summarize, the main contributions of this paper are concluded as follows:

- 1) Considering the nature of human aesthetic perception, we propose a disentangled training paradigm to study PFBP, which trains a *universal feature extractor* to capture the aesthetic commonality and a *personalized predictor* to adapt the aesthetic preferences of different users.
- 2) Based on the above training paradigm, we establish a *MetaFBP* framework, which adopts a novel learning-to-learn mechanism to optimize the personalized predictor. Specifically, we introduce a high-order predictor and optimize a meta-generator to twist the weights of the predictor quadratically for fast adaptation.
- 3) We build several PFBP benchmarks based on the existing FBP datasets. Extensive experiments on these benchmarks demonstrate the effectiveness and superiority of the proposed method to conventional meta-learning approaches. Our method can act as a strong baseline to study PFBP in the future works. Both benchmark datasets and source code are available at: <https://github.com/MetaVisionLab/MetaFBP>.

## 2 RELATED WORKS

**From Facial Beauty Prediction to Personalized Facial Beauty Prediction.** The goal of FBP is to train a model as smart as humans to estimate facial attractiveness. Conventional approaches [1, 3, 16, 46] tend to use geometric features or global appearance features (e.g., Color Histograms, Local Binary Pattern, Histogram of Oriented Gradients, Gabor Filters, etc.) to learn FBP. However, such handcraft features heavily depend on heuristic rules. Owing to the great success of deep learning [2, 19, 34], FBP can be easily optimized by Convolution Neural Network (CNNs) [13, 25–27, 47] in an end-to-end manner. However, most methods for FBP are designed to learn population aesthetics. PFBP is much less explored. In order to prepare data for learning personalized facial attractiveness preferences, Whitehill et al. [46] invited 8 volunteers to rate 1000 images. They trained regression models of facial beauty for each volunteer, and the experimental results indicated that personalized facial attractiveness preferences can be learnt by machine learning. Wang et al. [42] deemed that public aesthetic perception consisted of population aesthetics and personalized aesthetics. They decomposed the attractiveness score matrix into a low-rank matrix of population aesthetics and a matrix of personalized aesthetics, and used them to train regression models for learning population and personalized aesthetic jointly. Another study [35] focused on recommendation of personalized facial beauty for a large social website. Deep features of facial images extracted by a CNN are fed to collaborative filtering model. These works had validated that the subjective PFBP task can be solved by various machine

learning methods. However, none of them study PFBP under a few-shot learning setting which is much more applicable in real-world scenarios.

**Few-Shot Learning.** With the help of large-scale training data (e.g., ImageNet [11] and MS COCO [29]) and powerful computation resources, deep models have achieved great success [5, 8, 9, 18, 33]. However, deep models may fail to rapidly generalize to new tasks when given a few examples. To tackle this challenge, meta-learning [40] is proposed as a new learning paradigm. The purpose of meta-learning is to learn to solve the unseen new task using meta knowledge from various tasks instead of single task. Few-shot learning (FSL) [45], as an application of meta-learning, can learn from a small number of examples even without them (zero-shot learning [31, 43]). Researches of FSL have been greatly developed and can be categorized into many perspectives. Metric-based FSL [20, 37, 39, 41, 50] learns a representation space where similarities among samples are computed with a specific distance metric. Memory-based FSL [4, 30, 36] stores the learned knowledge as key-value pairs by using a memory component where new samples are considered as a query to match the most similar key. Optimization-based FSL is to use prior knowledge to search parameters which generalize better to novel tasks [17, 32, 38, 48]. Finn et al. [14] proposed a popular algorithm, MAML, to train the given neural network with a few gradient descent steps. To achieve this, MAML introduces two optimization loops for meta-learning, including an inner loop for task learning and an outer loop for training a meta-learner. The inner and outer loops are collaboratively optimized to find a meta-initialization that can be quickly adapted to different novel tasks. In this paper, we claim that PFBP is a more challenging task which requires a faster adaptation ability. To this end, for the first time, we upgrade the learning-to-learn mechanism with a high-order predictor and validate the significant superiority on PFBP.

**Personalized Image Aesthetics Assessment.** Personalized Image Aesthetics Assessment (PIAA) [12, 22, 49, 52, 53] aims to learn to assess the aesthetic quality (or score) of images by taking into account the users' aesthetic preferences. PIAA is a recent popular topic which is derived from the Generic Image Aesthetic Assessment (GIAA) [10, 51]. The PIAA are more related to our work as it learns personalized aesthetics for the image quality. Most PIAA works attempt to learn the individual aesthetic assessment by exploiting and transferring the learned knowledge from trained GIAA model [12, 22], or using extra supervision information [21, 53]. In this kind of task, the personalized aesthetics models are optimized to quickly adapt to a new user's aesthetic preference, and these PIAA models may fail to capture personalized aesthetics [52]. To this end, recent PIAA works [23, 44, 49, 52] based on meta-learning paradigms are proposed to tackle this problem. Although the promising PIAA performance is achieved, most methods still have complex training frameworks [21, 44, 52] that are not suitable for deployment in practice. Furthermore, existing methods mainly focus on FSL tasks with larger shots (10-shot and 100-shot), which means more labeled images are necessary for model fine-tuning. In this paper, we explore learning the personalized aesthetics for facial attractiveness with less supervision information in the standard FSL setting, leading to an urgent requirement on the fast adaptability using extremely-limited labeled examples.

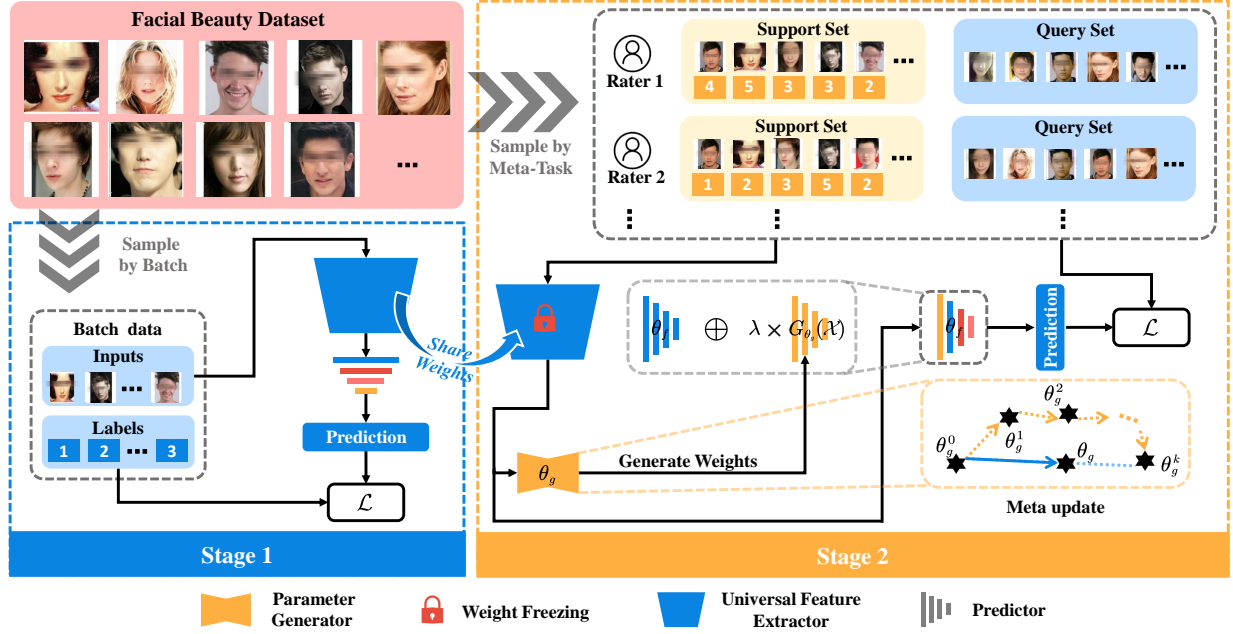


Figure 4: The illustration of the proposed MetaFBP framework. 1) Stage 1: Train a universal feature extractor. 2) Stage 2: Train a personalized high-order predictor, which is designed with a parameter generator  $\theta_g$ . We further optimize  $\theta_g$  via meta-learning to form a meta-generator so as to adapt to different new users given limited labeled images.

### 3 TASK FORMULATION

As mentioned above, a PFBP system can be arranged in the following manner: the system requires each user to label a few facial images that demonstrate their aesthetic preferences. The model is then fine-tuned on such limited labeled data for task adaptation. To meet this application manner, PFBP is modeled as a meta-learning paradigm in this paper. Specifically, each user corresponds to a meta-task. And all the meta-tasks are randomly divided into meta-train tasks and meta-test tasks. In each meta-task, the images are divided into a support set and a query set. In the meta-train tasks, the query set is a pseudo query set for meta-learning. In the meta-test tasks, the query set is used for performance evaluation without annotations. To provide sufficient aesthetic information in the support set for training, we urge each user to select the images from the gallery to traverse  $C$  different attractiveness scores with  $K$  samples per score, termed as “C-way K-shot”.

**Notations.** Figure 1 illustrates an example that formulates PFBP as meta-tasks. We represent the meta-train set as  $\mathcal{D}_{train} = \{\mathcal{D}_m\}_{m=1}^M$ , where  $M$  denotes the number of users and  $\mathcal{D}_m$  denotes the data rated by the  $m$ -th user. Then, a meta-task  $\mathcal{T}_m$  is sampled from  $\mathcal{D}_m = \{(\mathcal{X}_m, \mathcal{Y}_m)\}$  that contains the images  $\mathcal{X}_m$  and the corresponding beauty scores  $\mathcal{Y}_m$  rated by the  $m$ -th user. Each meta-task consists of a support set  $\mathcal{S}_m = \{(\mathcal{X}_m^s, \mathcal{Y}_m^s)\}$  and a query set  $\mathcal{Q}_m = \{(\mathcal{X}_m^q, \mathcal{Y}_m^q)\}$ , where  $\mathcal{X}_m^s, \mathcal{X}_m^q \in \mathcal{X}_m$  and  $\mathcal{Y}_m^s, \mathcal{Y}_m^q \in \mathcal{Y}_m$ . The support set  $\mathcal{S}_m$  and query set  $\mathcal{Q}_m$  are constructed by randomly selecting  $N_s$  and  $N_q$  samples from the subset  $\mathcal{D}_m$  for each attractiveness score without overlapping. Similarly, the meta-test set is built in the same way, where the support set is used for model fine-tuning, but leaving the unlabeled query set to evaluate the adaptation performance of the model fine-tuned by the support set.

**Training objective.** Since our method is strongly correlated with the optimization-based meta-learning methods [14], we only present the training pipeline of the optimization-based methods for the PFBP task. For each meta-task/episode, we first update a model  $F(\cdot)$  on the support set  $\mathcal{S}_m$  by calculating the regression loss, aka the mean squared error (MSE) loss  $\mathcal{L}_{mse}(F(\mathcal{X}_m^s), \mathcal{Y}_m^s)$ , and then use the updated model to predict the beauty scores of the query images  $\mathcal{X}_m^q$ . The predicted scores can be formulated as follows:

$$\hat{\mathcal{Y}}_m^q = F(\mathcal{X}_m^q | \nabla \mathcal{L}_{mse}(F(\mathcal{X}_m^s), \mathcal{Y}_m^s)). \quad (1)$$

Subsequently, the training objective of each meta-task is to minimize the regression loss between the predicted scores and the corresponding ground-truth rating labels:

$$\min_F \mathcal{L}_{mse}(\hat{\mathcal{Y}}_m^q, \mathcal{Y}_m^q). \quad (2)$$

### 4 METHOD

The human aesthetic preference can be disentangled into a commonality part and a personality part. The former represents the consistent judgement of the majority while the latter represents the individual variations from the majority. To meet this prior knowledge, we decompose the network architecture into a universal feature extractor to capture the aesthetic commonality and a predictor to capture the aesthetic personality by shifting the decision boundary for task adaptation. To enhance task adaptation, we introduce a high-order predictor which updates the predictor by using a shallow parameter-generator network. We further optimize the generator into a meta-generator via meta-learning. These two components (commonality vs. personality) are optimized independently with different optimization objectives. The whole training process is illustrated in Figure 4.

#### 4.1 Stage 1: Universal Feature Extractor

In the first stage, we aim to learn a universal feature extractor to capture aesthetic commonality by training a common FBP model, which is composed of a feature extractor  $E_{\theta_e}$  and a predictor  $F_{\theta_f}$ . This model is trained identical to the common FBP training manner. Since the labels used in this stage are required to represent aesthetic commonality, we choose *mode* rating to ensemble the scores from all users, that is, the most-popular score in the rating distribution is exploited as the true label of each image. Take an image  $x$  as an example, its rating distribution is represented as  $\{y_1, \dots, y_m, \dots, y_M\}$  where  $y_m$  is the attractiveness score rated by the  $m$ -th user. The label representing the aesthetic commonality is defined as:

$$\bar{y} = \arg \max_c \sum_{m=1}^M \delta(y_m = c), \quad (3)$$

where  $c$  ranges from 1 to  $C$ . *mode* rating is considered more representative than *mean* rating to capture aesthetic commonality, since it does not take into account the opinions of minorities. Using  $\bar{y}$  as the supervision signal, the training goal can be formulated as:

$$\min_{\theta_e, \theta_f} \sum_{(x, \bar{y}) \in \mathcal{D}_{train}} \mathcal{L}(F_{\theta_f}(E_{\theta_e}(x)), \bar{y}), \quad (4)$$

where  $\mathcal{L}$  is a prediction loss. Upon completion of the training stage for the common FBP model, the predictor  $F_{\theta_f}$  is discarded, and only the universal feature extractor  $E_{\theta_e}$  is retained. It is worth noting that in the subsequent stage, the weights of the universal feature extractor are fixed, with the goal of maintaining the knowledge of aesthetic commonality across different meta-tasks.

#### 4.2 Stage 2: Personalized High-Order Predictor

Based on the commonality-aware feature extractor in the first stage, we need a personalized predictor to meet various user preferences by shifting the decision boundary adaptively. One straightforward approach is to use the conventional meta-learning methods like MAML [14] to optimize the predictor. However, PFBP is more challenging than previous image recognition tasks, which requires a faster adaptation ability, while the conventional meta-learning methods usually result in slow adaptation or overfitting the tiny support sets. As an alternative, we propose a high-order predictor that is dynamically twisted conditioned on the inputs, by using a shallow parameter-generator. We optimize the generator via a gradient-based meta-learning method to achieve a meta-generator which can adapt the meta-tasks more quickly. See Figure 3 to compare with conventional meta-learning mechanism. The implementation details are shown in Figure 4 and Algorithm 1.

**High-Order Predictor.** In this paper, the predictor is implemented as a fully-connected (FC) layer with weights  $\theta_f$ . To enhance adaptation ability, a high-order predictor is encouraged here. Specifically, the weights  $\theta_f$  in the high-order predictor can be twisted adaptively in test-time, which is formulated as:

$$\theta_f = \theta_f + \lambda G_{\theta_g}(X), \quad (5)$$

where  $G$  is a shallow parameter generator conditioned on the input features  $X$  provided by the feature extractor.  $\theta_g$  is the weight of the generator.  $\lambda$  is a hyper-parameter that represents the adaptation strength. In practice, the parameter generator is implemented as

---

#### Algorithm 1: Learning-to-learn high-order predictor

---

```

1 Require: Training data  $\mathcal{D}_{train} = \{\mathcal{D}_m\}_{m=1}^M$ ,  $k$ -step in the
   inner loop, training iteration  $I$ 
2 Randomly initialize weights  $\theta_f, \theta_g$ ;
3 for  $i \leftarrow 1$  to  $I$  do
4   Sample a meta-task  $\mathcal{T}_m$  from  $\mathcal{D}_m$ ,  $m \in [1, M]$ ;
5   Acquire the support set  $\mathcal{S}_m \sim \mathcal{T}_m$ ;
6    $\theta'_g \leftarrow \theta_g$ ;
7   repeat
8     Update  $\theta'_g$  on  $\mathcal{S}_m$  using Equation 6;
9   until  $k$  times;
10  Acquire the query set  $\mathcal{Q}_m \sim \mathcal{T}_m$ ;
11  Update  $\theta_f, \theta_g$  on  $\mathcal{Q}_m$  using Equation 7;
12 end

```

---

a multi-layer-perceptron with the structure of FC-ReLU-FC. Compared with the first-order predictor, the high-order predictor has a much higher freedom for task adaptation.

**Meta-Generator.** Based on the design of high-order predictor, we aim to further optimize the parameter-generator into a meta-generator via meta-learning. Roughly speaking, in the inner loop, we conduct  $k$ -step adaptation to optimize  $\theta_g$  on the support set  $\mathcal{S}_m$  to achieve a ghosted  $\theta_g$ , termed  $\theta'_g$ . In the outer loop, we conduct meta-update to  $\theta_g$  on the query set  $\mathcal{Q}_m$  by using  $\theta'_g$ . In detail, the prediction of the high-order predictor on the support set can be represented as  $F_{\theta_f \circ \theta_g}(X_m^s)$ . The weights  $\theta_g$  are updated based on the gradients of the model on the support set  $\mathcal{S}_m$ :

$$\theta'_g \leftarrow \theta_g - \underbrace{\alpha \nabla_{\theta_g} \mathcal{L}(F_{\theta_f \circ \theta_g}(X_m^s), \mathcal{Y}_m^s)}_{k\text{-step}}, \quad (6)$$

where  $\alpha$  is a hyper-parameter representing the step size of inner update. The process in Equation 6 is repeated  $k$  times to obtain a more task-oriented gradient that makes the adaptation more thoroughly. We then calculate the quadratic gradients of the updated predictor on the query set to update  $\theta_g$  by taking  $\theta'_g$  as a bridge:

$$[\theta_f, \theta_g] \leftarrow [\theta_f, \theta_g] - \beta \nabla_{\theta_f, \theta_g} \mathcal{L}(F_{\theta_f \circ \theta'_g}(X_m^q), \mathcal{Y}_m^q), \quad (7)$$

where  $\beta$  is a hyper-parameter denoting the step size of outer loop. Note that we update the generator based on its initial weights to obtain a generalizable initial weights. Different from  $\theta_g, \theta_f$  is optimized in a standard training manner.

## 5 EXPERIMENTS

### 5.1 Experimental Benchmarks

We construct three PFBP benchmarks varying from small to large scales, based on the public FBP datasets that provide beauty scores annotated by multiple raters. (1) **PFBP-SCUT5500** is collected from SCUT-FBP5500 [24], which consists of 5,500 face images, each labeled by 60 users. The images vary widely in terms of characteristics such as gender and ethnicity, making it difficult to predict esthetic preferences. (2) **PFBP-SCUT500** is collected from SCUT-FBP500

**Table 1: The details of dataset split.**

Item	PFBP-SCUT5500			
	Total	Train Set	Validation Set	Test Set
Number of images	500	300	50	150
Number of users	60	30	10	20
Total annotations	12,500	9,000	500	3,000

Item	PFBP-SCUT500			
	Total	Train Set	Validation Set	Test Set
Number of images	5,500	3,000	500	2,000
Number of users	60	30	10	20
Total annotations	135,000	90,000	5,000	40,000

Item	PFBP-US10K			
	Total	Train Set	Validation Set	Test Set
Number of images	2,222	1,111	667	444
Number of users	12	6	2	4
Total annotations	9,771	6,663	1,333	1,775

[47], which contains 500 facial images from the Asian female population rated by 75 volunteers. (3) **PFBP-US10K** is sampled from 10K US Adult Faces dataset [3], consisting 2,222 facial images from Caucasian population rated by 12 users. All the above datasets are annotated at a beauty scale of  $\{1, 2, 3, 4, 5\}$ , where the higher beauty score represents the more attractive face.

**Dataset Split.** We split each dataset into train, validation and test sets in a 6:3:1 ratio based on the users who provided the annotations. Additionally, we ensure that the images in each split are distinct. Note that some users provided empty ratings for the extremely high or low beauty score, making it difficult to sample meta-tasks from these users. As a result, we exclude such users from the split. The details of dataset split are listed in Table 1.

**Evaluation Protocols.** The model that performed the best on the validation set is chosen for evaluation. We use Pearson correlation (PC), mean absolute error (MAE), and root mean squared error (RMSE) to measure the regression performance of our method. A higher PC, smaller MAE and RMSE indicate the better performance achieved by the model on PFBP task.

## 5.2 Experimental Details

**Setup.** We follow the meta-training and meta-testing settings from few-shot learning tasks to conduct experiments. Specifically, we perform 5-way K-shot regression meta-tasks on PFBP-SCUT5500 and PFBP-US10K. However, due to a large number of empty annotations for certain categories in PFBP-SCUT500, we rearrange the score labels by reducing the beauty category number from 5 to 3 via score mapping:  $\{1, 2\} \rightarrow 1$ ,  $\{3\} \rightarrow 2$ ,  $\{4, 5\} \rightarrow 3$ . Afterwards, we conduct 3-way K-shot regression meta-tasks on the PFBP-SCUT500.

**Cyclically Re-sampling Strategy.** When performing each meta-task, we need to select  $N_s + N_q$  images per category to create the support set and the query set. However, user ratings are usually imbalanced, with a few samples receiving scores of 1 and 5, while thousands receive a score of 2. This can lead to an extreme situation where the number of images in the minority categories is insufficient to create a meta-task. To solve this problem, we devise a Cyclically Re-sampling Strategy for the minority categories. Assuming  $N_c$  denotes the sample number of the  $c$ -th category rated by the  $m$ -th user, the sampling strategy is defined as follows:

**Case 1:** If  $N_c = 1$ , we duplicate the single sample  $N_s$  times to form the  $c$ -th category in the support set, and the  $c$ -th category in the query set will be left empty.

**Case 2:** If  $1 < N_c \leq N_s$ , we first randomly select  $N_c - 1$  samples from the  $c$ -th category to form the support set by duplicated sampling  $K/(N_c - 1)$  times, and the remaining one is duplicated sampling  $N_q$  times to form the query set.

**Case 3:** If  $N_s < N_c \leq N_s + N_q$ , we first randomly select  $N_s$  samples from the  $c$ -th category to form the support set, and the remaining samples are duplicated sampling  $N_q/(N_c - N_s)$  times to form the query set.

**Case 4:** If  $N_c > N_s + N_q$ , we first randomly select  $N_s$  samples from the  $c$ -th category to form the support set, and then randomly select  $N_q$  samples from the remaining ones to form the query set.

**Implementation Details.** Our experiments are implemented on Pytorch platform and runs on a NVIDIA RTX3090 GPU. For all experiments, we use ResNet-18 [15] as the network backbone, which is initialized by the ImageNet pre-trained model [11]. Before training, we simply apply several augmentation techniques to preprocess images, including random crop and random horizontal flipping. During the first training stage, we train the universal feature extractor by using cross-entropy loss and SGD optimizer with batchsize of 64, maximum epochs of 100, and a learning rate of 0.001 stepped down by half per 20 epochs. In the second stage, we freeze the weights of the universal feature extractor, and develop a high-order predictor by implementing a meta-generator that is a MLP with structure of FC-ReLU-FC. The hyper-parameter of adaptation strength  $\lambda$  is set to 0.01. The inner-update step size  $\alpha$ , outer-update step size  $\beta$  and the number of  $k$ -step are set to 0.01, 0.001 and 10, respectively. And we sample 40,000 and 400 meta-tasks from the training set and the testing set for meta-training and meta-testing, respectively. Note that the shot number of the support set is kept consistent in both the meta-training and meta-testing phases, if without additional explanation. And the shot number of the query set is set to 15 by default.

**Strong Baselines.** As the first time applying meta-learning to formulate PFBP, there are currently no experimental results on these new benchmarks. To demonstrate the effectiveness of our proposed method, we also implement several strong baselines on these PFBP benchmarks, including: 1) **Base-commonFBP**: In line with conventional training methods [24], we developed a common FBP model with the same architecture as our model. The common FBP model can represent the aesthetic commonality, which is assumed to be correlated with user preferences to some extent. We then evaluate its effectiveness on the PFBP task using the same meta-testing manner as our approach. 2) **Base-MAML**: MAML [14] is a popular meta-learning approach to address few-shot learning tasks. To highlight the advantage of our method, we also implement MAML on the PFBP task. For a fair comparison, MAML is implemented using the same task formulation, architecture, and hyper-parameter settings (e.g.,  $\alpha, \beta, \gamma$ ) as our method.

**Other Related Methods.** In order to provide a comprehensive evaluation of the proposed method, we also compare it with other state-of-the-art methods on our PFBP task. Specifically, we implement **FSL** methods, i.e., ProtoNet [37] and MTL [38], and

**Table 2: 5-way K-shot regression results on PFBP-SCUT5500 benchmark. The same number of shots is kept during both meta-training and meta-testing phases. The best and second-best results are marked by bold and underline, respectively. Same representation in the following tables.**

Type	Method	PC	1 Shot MAE	RMSE	PC	5 Shot MAE	RMSE	PC	10 Shot MAE	RMSE
Baseline	Base-CommonFBP	0.6827	0.8668	1.1135	0.7812	0.7088	0.9044	0.7992	0.6826	0.8481
	Base-MAML [14]	0.7549	0.8480	1.1268	0.7837	0.7766	1.0245	0.7862	0.7906	1.0249
FSL	ProtoNet [37]	0.7816	0.7053	0.8620	0.7969	0.6838	0.8450	0.7980	0.6932	0.8875
	MTL [38]	0.7228	0.8593	1.0972	0.7350	0.9040	1.1832	0.7277	0.8924	1.1375
PIAA	BLG-PIAA [52]	0.7927	0.6850	0.8426	0.7683	0.7853	1.1154	0.7705	0.7995	1.0434
Ours	MetaFBP-R	<u>0.8037</u>	<u>0.6780</u>	<u>0.8365</u>	<u>0.8050</u>	<u>0.6727</u>	<u>0.8318</u>	<u>0.8098</u>	<u>0.6631</u>	<u>0.8208</u>
	MetaFBP-T	<b>0.8067</b>	<b>0.6701</b>	<b>0.8274</b>	<b>0.8061</b>	<b>0.6716</b>	<b>0.8282</b>	<b>0.8125</b>	<b>0.6572</b>	<b>0.8147</b>

**Table 3: 3-way K-shot regression results on PFBP-SCUT500 benchmark.**

Type	Method	PC	1 Shot MAE	RMSE	PC	5 Shot MAE	RMSE	PC	10 Shot MAE	RMSE
Baseline	Base-CommonFBP	0.5324	0.5335	0.6658	0.7206	0.4379	0.5491	0.7525	0.4444	0.5342
	Base-MAML [14]	0.7074	0.4244	0.5659	0.7688	0.3915	0.5040	0.7708	0.3840	0.4996
FSL	ProtoNet [37]	0.6269	0.5279	0.5920	0.7693	0.3907	0.5081	0.7694	0.3865	0.5175
	MTL [38]	0.6706	0.5196	0.7156	0.6739	0.5192	0.7152	0.6645	0.5157	0.7121
PIAA	BLG-PIAA [52]	0.7139	0.3995	0.5439	0.7369	0.4029	0.5372	0.7577	0.3995	0.5238
Ours	MetaFBP-R	<b>0.7478</b>	<b>0.3840</b>	<b>0.5335</b>	<b>0.7769</b>	<b>0.3720</b>	<u>0.5026</u>	<u>0.7729</u>	<u>0.3772</u>	<u>0.4960</u>
	MetaFBP-T	<u>0.7393</u>	<u>0.3947</u>	<u>0.5378</u>	<u>0.7738</u>	<u>0.3775</u>	<b>0.5025</b>	<b>0.7787</b>	<b>0.3746</b>	<b>0.4911</b>

**Table 4: 5-way K-shot regression results on PFBP-US10K benchmark.**

Type	Method	PC	1 Shot MAE	RMSE	PC	5 Shot MAE	RMSE	PC	10 Shot MAE	RMSE
Baseline	Base-CommonFBP	0.2640	1.2355	1.5295	0.3871	1.1094	1.3507	0.4742	1.0473	1.2491
	Base-MAML [14]	0.4028	1.2719	1.6584	0.4286	1.2425	1.5975	0.4570	1.2324	1.5621
FSL	ProtoNet [37]	0.3094	1.2147	1.5943	0.4877	1.0260	1.2683	0.4906	1.0244	1.2389
	MTL [38]	0.4678	1.0311	1.3136	0.4642	1.0322	1.3137	0.4610	1.0594	1.3774
PIAA	BLG-PIAA [52]	0.4691	1.0281	1.3066	0.4615	1.0476	1.3599	0.4731	1.0502	1.3339
Ours	MetaFBP-R	<u>0.5067</u>	<u>0.9970</u>	<u>1.2356</u>	<u>0.4968</u>	<u>1.0121</u>	<b>1.2344</b>	<b>0.5004</b>	<b>1.0115</b>	<b>1.2290</b>
	MetaFBP-T	<b>0.5109</b>	<b>0.9968</b>	<b>1.2182</b>	<b>0.5007</b>	<b>1.0086</b>	<u>1.2425</u>	<u>0.4991</u>	<u>1.0128</u>	<u>1.2322</u>

a recent **PIAA** method, i.e., BLG-PIAA [52] on PFBP task. Since ProtoNet is originally designed for few-shot classification tasks, we modify it for PFBP task by calculating the expectation score of the output distribution as the final prediction result.

### 5.3 Experimental Results

To further investigate the impact of different parameter updating manners on the high-order predictor, we implement two different methods, known as parameter-tuning and parameter-rebirth. Parameter tuning aims to modulate the parameters of the predictor by generating dynamic residuals to add to the original parameters. The operation of parameter tuning is illustrated in Equation 5. Unlike parameter tuning, parameter rebirth discards the original parameters and generates the new parameters by the parameter generator  $G_{\theta_g}$  conditioned on the input features  $\mathcal{X}$ . The operation of parameter rebirth can be formulated as:  $\theta_f = G_{\theta_g}(\mathcal{X})$ . For simplicity, our method implemented with parameter tuning and parameter rebirth are termed as **MetaFBP-T** and **MetaFBP-R**, respectively.

**Comparison with Strong Baselines.** Our method is compared with the strong baselines (i.e., Base-commonFBP and Base-MAML) to stress its effectiveness on PFBP. The comparison results on PFBP-SCUT5500, PFBP-SCUT500 and PFBP-US10K benchmarks in terms

of PC, MAE and RMSE are reported in Table 2, Table 3 and Table 4, respectively. From these tables, we can observe that our method almost surpasses all the strong baselines with a much higher PC and smaller MSE, RMSE over all the benchmarks, in terms of different K-shot settings. For the most challenging 1-shot setting, our method (MetaFBP-R and MetaFBP-T) both achieve a great PC improvements of more than 4% and 3% on the PFBP-SCUT5500 and PFBP-SCUT500 benchmarks, respectively, compared to the baselines. Moreover, our method demonstrates a more significant performance improvement over Base-MAML on the user-less PFBP-US10K benchmark, with an improvement in PC more than 10%. This result highlights the ability of our method to adapt to new tasks even when training data is limited. Furthermore, as the number of training shots increases, the performance of our method improves correspondingly, with the most significant improvement observed in the 1-shot setting. In practically, the improvement in 1-shot setting is particularly significant in real-world scenarios as it allows for a more convenient user experience with fewer required ratings.

**Comparison with Other Related Methods.** From Table 2–4, we can observe that our method achieves the state-of-the-art results even compared with competitive methods, including PIAA and FSL methods, among all benchmarks, which demonstrates the effectiveness of our metaFBP method on PFBP task.

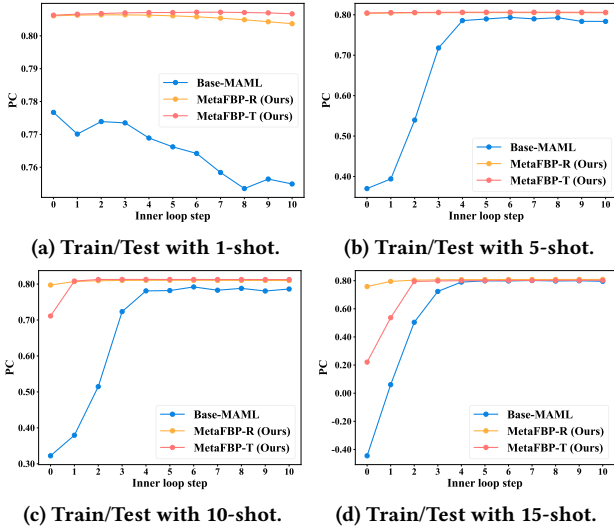


**Table 5: Ablation study of different K-shot settings during training and testing phases on PFBP-SCUT5500 dataset.**

Method	Training Phase	Testing Phase		
		1 shot	5 shot	10 shot
Base-MAML	1 shot	0.7549	0.7690	0.7809
MetaFBP-T		<b>0.8067</b>	<b>0.8050</b>	<b>0.8104</b>
Base-MAML	5 shot	0.7495	0.7837	0.7898
MetaFBP-T		<b>0.8033</b>	<b>0.8061</b>	<b>0.8118</b>
Base-MAML	10 shot	0.7309	0.7758	0.7862
MetaFBP-T		<b>0.7914</b>	<b>0.8057</b>	<b>0.8125</b>

**Table 6: Ablation study of the adaptation strength  $\lambda$ .**

$\lambda$	1 shot	5 shot	Avg
1	0.5915	0.2026	0.3971
0.1	0.8059	0.7231	0.7645
0.01	<b>0.8067</b>	<b>0.8061</b>	<b>0.8064</b>
0.001	0.6846	0.7575	0.7211
0.0001	0.4932	0.6932	0.5932

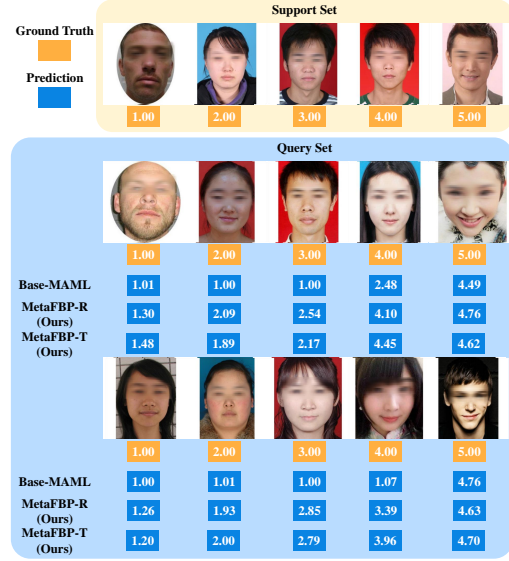
**Figure 5: Pearson Correlation (PC) with respect to  $k$ -step in the inner loop of different models under different K-shot settings on PFBP-SCUT5500 benchmark.**

## 5.4 Ablation Study

### Different K-shot settings during training and testing phases.

To further investigate the effectiveness of our method, we conduct extensive experiments on PFBP-SCUT5500, which train a model with specific shot number and test the model with different shot settings. We only report the results of PC, which are listed in Table 5. We can find that for a model trained with specific shots, it can be improved with the increasing K-shot of support set. Our method (trained with 1-shot and tested with 1-shot) still outperforms the Base-MAML (trained with 1-shot and tested with 10-shot). It again demonstrates that our method has faster adaptation ability than MAML, even using less labeled data during fine-tuning.

**Exploring Adaptation Strength  $\lambda$ .** It shows in Equation 5 that the adaptation strength  $\lambda$  controls the adaptation magnitude. We investigate the effectiveness of different  $\lambda$  on PFBP-SCUT5500 in Table 6, from which we can observe that neither larger or smaller  $\lambda$  can improve performance. Too large  $\lambda$  may destroy the weights

**Figure 6: The prediction results of a specific user provided by different models trained with PFBP-SCUT500 dataset.**

of the predictor so that causes drastic performance degradation. The smaller  $\lambda$  can reduce the risk of over-fitting. However, too small  $\lambda$  will make the high-order predictor finally trash into a plain predictor. Therefore, we set  $\lambda$  to a normal value of 0.01.

**Visualization.** An intuitive way to visualize the fast adaption of our method is shown in Figure 5. It can be seen that our method keeps the best performance with less variation on 1-shot (Figure 5a) and 5-shot (Figure 5b). For the 10-shot (Figure 5c) and 15-shot (Figure 5d) settings, our method earlier reaches the top max PC compared with MAML, which shows the proposed method can solve the slow adaptation and overfitting problems in conventional meta-learning methods. We also plot the prediction result for the most challenging 1-shot task on PFBP-SCUT500 benchmark. Figure 6 reveals that the Base-MAML model lacks ability to capture individual aesthetic preferences because it frequently assigns low scores to facial images, regardless of their actual differences. Conversely, our method can produce varying scores for different images, resulting in a higher correlation with the true labels.

## 6 CONCLUSION

In this paper, we delve into Personalized Facial Beauty Prediction (PFBP). We model PFBP into a Few-Shot Learning (FSL) task and discuss its different challenge from conventional FSL task. We claim that PFBP requires a faster adaptation ability considering its user-adaptive characteristic, while the conventional meta-learning methods to solve FSL are usually trapped into slow adaptation or overfitting the tiny support set. To solve this problem, we develop a learning-to-learn mechanism into a high-order predictor for fast adaptation. Extensive quantitative and qualitative experiments demonstrate the effectiveness of the proposed method.

## ACKNOWLEDGMENTS

This work was supported by the Fujian Provincial Natural Science Foundation (No. 2022J05135), the University-Industry Cooperation Project of Fujian Provincial Department of Science and Technology (No. 2020H6005), and the National Natural Science Foundation of China (No. U21A20471).



## REFERENCES

- [1] Parham Aarabi, Dominic Hughes, Keyvan Mohajer, and Majid Emami. 2001. The automatic measurement of facial beauty. In *IEEE SMC*, Vol. 4. 2644–2647.
- [2] Ossama Abdel-Hamid, Abdel-rahman Mohamed, Hui Jiang, Li Deng, Gerald Penn, and Dong Yu. 2014. Convolutional neural networks for speech recognition. *IEEE/ACM Trans. Audio Speech Lang.* 22, 10 (2014), 1533–1545.
- [3] Andrea Bottino and Aldo Laurentini. 2012. The intrinsic dimensionality of attractiveness: A study in face profiles. In *CIARP*. 59–66.
- [4] Qi Cai, Yingwei Pan, Ting Yao, Chenggang Yan, and Tao Mei. 2018. Memory matching networks for one-shot image recognition. In *CVPR*. 4080–4088.
- [5] Binbin Chen, Weijie Chen, Shicai Yang, Yunyi Xuan, Jie Song, Di Xie, Shiliang Pu, Mingli Song, and Yueting Zhuang. 2022. Label Matching Semi-Supervised Object Detection. In *CVPR*. 14381–14390.
- [6] Fangmei Chen, Xihua Xiao, and David Zhang. 2016. Data-driven facial beauty analysis: prediction, retrieval and manipulation. *IEEE Trans. Affect. Comput.* 9, 2 (2016), 205–216.
- [7] Wang Chen, Peizhen Chen, Weijie Chen, and LuoJun Lin. 2023. Customized Automatic Face Beautification. In *ICASSP*. 1–5.
- [8] Weijie Chen, Shiliang Pu, Di Xie, Shicai Yang, Yilu Guo, and LuoJun Lin. 2020. Unsupervised image classification for deep representation learning. In *ECCV*. 430–446.
- [9] Weijie Chen, Di Xie, Yuan Zhang, and Shiliang Pu. 2019. All you need is a few shifts: Designing efficient convolutional neural networks for image classification. In *CVPR*. 7241–7250.
- [10] Chaoran Cui, Huihui Liu, Tao Lian, Liqiang Nie, Lei Zhu, and Yilong Yin. 2018. Distribution-oriented aesthetics assessment with semantic-aware hybrid network. *TMM* 21, 5 (2018), 1209–1220.
- [11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *CVPR*. 248–255.
- [12] Xiang Deng, Chaoran Cui, Huidi Fang, Xiushan Nie, and Yilong Yin. 2017. Personalized image aesthetics assessment. In *ACM CIKM*. 2043–2046.
- [13] Yang-Yu Fan, Shu Liu, Bo Li, Zhe Guo, Ashok Samal, Jun Wan, and Stan Z Li. 2017. Label distribution-based facial attractiveness computation by deep residual learning. *TMM* 20, 8 (2017), 2196–2208.
- [14] Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*. 1126–1135.
- [15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*. 770–778.
- [16] Amit Kagian, Gideon Dror, Tommer Leyvand, Daniel Cohen-Or, and Eytan Ruppin. 2007. A humanlike predictor of facial attractiveness. In *NeurIPS*. 649–656.
- [17] Konstantinos Kalais and Sotirios Chatzis. 2022. Stochastic Deep Networks with Linear Competing Units for Model-Agnostic Meta-Learning. In *ICML*. 10586–10597.
- [18] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. 2023. Segment anything. *arXiv preprint arXiv:2304.02643* (2023).
- [19] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *NeurIPS*, Vol. 25.
- [20] Hongyang Li, David Eigen, Samuel Dodge, Matthew Zeiler, and Xiaogang Wang. 2019. Finding task-relevant features for few-shot learning by category traversal. In *CVPR*. 1–10.
- [21] Leida Li, Hancheng Zhu, Sicheng Zhao, Guiguang Ding, Hongyan Jiang, and Allen Tan. 2019. Personality driven multi-task learning for image aesthetic assessment. In *ICME*. 430–435.
- [22] Leida Li, Hancheng Zhu, Sicheng Zhao, Guiguang Ding, and Weisi Lin. 2020. Personality-assisted multi-task learning for generic and personalized image aesthetics assessment. *TIP* 29 (2020), 3898–3910.
- [23] Yaohui Li, Yuzhe Yang, Huaxiong Li, Haoxing Chen, Liwu Xu, Leida Li, Yaqian Li, and Yandong Guo. 2022. Transductive aesthetic preference propagation for personalized image aesthetics assessment. In *ACM MM*. 896–904.
- [24] Lingyu Liang, LuoJun Lin, Lianwen Jin, Duorui Xie, and Mengru Li. 2018. SCUT-FBP5500: A diverse benchmark dataset for multi-paradigm facial beauty prediction. In *ICPR*. 1598–1603.
- [25] Lingyu Liang, Duorui Xie, Lianwen Jin, Jie Xu, Mengru Li, and LuoJun Lin. 2017. Region-aware scattering convolution networks for facial beauty prediction. In *ICIP*. 2861–2865.
- [26] LuoJun Lin, Lingyu Liang, and Lianwen Jin. 2018. R2-resnext: A resnext-based regression model with relative ranking for facial beauty prediction. In *ICPR*. 85–90.
- [27] LuoJun Lin, Lingyu Liang, and Lianwen Jin. 2019. Regression Guided by Relative Ranking Using Convolutional Neural Network (R3CNN) for Facial Beauty Prediction. *IEEE Trans. on Affect. Comput.* 13, 1 (2019), 122–134.
- [28] LuoJun Lin, Lingyu Liang, Lianwen Jin, and Weijie Chen. 2019. Attribute-Aware Convolutional Neural Networks for Facial Beauty Prediction. In *IJCAI*. 847–853.
- [29] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *ECCV*. 740–755.
- [30] Tsendsuren Munkhdalai and Hong Yu. 2017. Meta networks. In *ICML*. 2554–2563.
- [31] Sanath Narayan, Akshita Gupta, Fahad Shahbaz Khan, Cees GM Snoek, and Ling Shao. 2020. Latent embedding feedback and discriminative features for zero-shot classification. In *ECCV*. 479–495.
- [32] Alex Nichol, Joshua Achiam, and John Schulman. 2018. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999* (2018).
- [33] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *ICML*. 8748–8763.
- [34] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2016. Faster R-CNN: towards real-time object detection with region proposal networks. *TPAMI* 39, 6 (2016), 1137–1149.
- [35] Rasmus Rothe, Radu Timofte, and Luc Van Gool. 2016. Some like it hot-visual guidance for preference prediction. In *CVPR*. 5553–5561.
- [36] Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy Lillicrap. 2016. Meta-learning with memory-augmented neural networks. In *ICML*. 1842–1850.
- [37] Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. *NeurIPS* 30 (2017).
- [38] Qianru Sun, Yaoyao Liu, Tat-Seng Chua, and Bernt Schiele. 2019. Meta-transfer learning for few-shot learning. In *CVPR*. 403–412.
- [39] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. 2018. Learning to compare: Relation network for few-shot learning. In *CVPR*. 1199–1208.
- [40] Sebastian Thrun and Lorien Pratt. 1998. Learning to learn: Introduction and overview. In *Learning to learn*. Springer, 3–17.
- [41] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. 2016. Matching networks for one shot learning. *NeurIPS* 29 (2016), 3630–3638.
- [42] Shaobiao Wang, Lu Fang, and Juyong Zhang. 2014. [Demo paper] exploring attractive faces: General versus personal preferences. In *ICMEW*. 1–2.
- [43] Wenlin Wang, Yunchen Pu, Vinay Verma, Kai Fan, Yizhe Zhang, Changyou Chen, Piyush Rai, and Lawrence Carin. 2018. Zero-shot learning via class-conditioned deep generative models. In *AAAI*, Vol. 32.
- [44] Weining Wang, Junjie Su, Lemin Li, Xiangmin Xu, and Jiebo Luo. 2019. Meta-learning perspective for personalized image aesthetics assessment. In *ICIP*. 1875–1879.
- [45] Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. 2020. Generalizing from a few examples: A survey on few-shot learning. *ACM Comput. Surv.* 53, 3 (2020), 1–34.
- [46] Jacob Whitehill and Javier R Movellan. 2008. Personalized facial attractiveness prediction. In *FG IEEE*, 1–7.
- [47] Duorui Xie, Lingyu Liang, Lianwen Jin, Jie Xu, and Mengru Li. 2015. Scut-fbp: A benchmark dataset for facial beauty perception. In *SMC*. 1821–1826.
- [48] Hansi Yang and James Kwok. 2022. Efficient Variance Reduction for Meta-Learning. In *ICML*. 25070–25095.
- [49] Yuzhe Yang, Liwu Xu, Leida Li, Nan Qie, Yaqian Li, Peng Zhang, and Yandong Guo. 2022. Personalized image aesthetics assessment with rich attributes. In *CVPR*. 19861–19869.
- [50] Sung Whan Yoon, Jun Seo, and Jaekyun Moon. 2019. Tapnet: Neural network augmented with task-adaptive projection for few-shot learning. In *ICML*. 7115–7123.
- [51] Xiaodan Zhang, Xinbo Gao, Wen Lu, and Lihuo He. 2019. A gated peripheral-foveal convolutional neural network for unified image aesthetic prediction. *TMM* 21, 11 (2019), 2815–2826.
- [52] Hancheng Zhu, Leida Li, Jinjian Wu, Sicheng Zhao, Guiguang Ding, and Guangming Shi. 2020. Personalized image aesthetics assessment via meta-learning with bilevel gradient optimization. *IEEE Trans. on Cyber.* 52, 3 (2020), 1798–1811.
- [53] Hancheng Zhu, Yong Zhou, Leida Li, Yaqian Li, and Yandong Guo. 2023. Learning Personalized Image Aesthetics From Subjective and Objective Attributes. *TMM* 25 (2023), 179–190.