ABSTRACT

Many ubiquitous computing projects have addressed health and wellness behaviors such as healthy eating. Healthy meal recommendations have the potential to help individuals prevent or manage conditions such as diabetes and obesity. However, learning people’s food preferences and making healthy recommendations that appeal to their palate is challenging. Existing approaches either only learn high-level preferences or require a prolonged learning period. We propose Yum-me, a personalized healthy-meal recommender system designed to meet individuals’ health goals, dietary restrictions, and fine-grained food preferences. Marrying ideas from user preference learning and healthy eating promotion, Yum-me enables a simple and accurate food preference profiling procedure via an image-based online learning framework, and projects the learned profile into the domain of healthy food options to find ones that will appeal to the user. We present the design and implementation of Yum-me, and further discuss the most critical component of it: FoodDist, a state-of-the-art food image analysis model. We demonstrate FoodDist’s superior performance through careful benchmarking, and discuss its applicability across a wide array of dietary applications. We validate the feasibility and effectiveness of Yum-me through a 60-person user study, in which Yum-me improves the recommendation acceptance rate by 42.63% over the traditional food preference survey.

ACM Classification Keywords

H.5.m. Information Interfaces and Presentation (e.g. HCl): Miscellaneous

Author Keywords

Healthy meal recommendation; personalization; visual interface; food preferences

INTRODUCTION

Healthy eating plays a critical role in our daily well-being and is indispensable in preventing and managing conditions such as diabetes, high blood pressure, cancer, mental illnesses, asthma, etc [36, 20, 9]. In particular, for children and young people, the adoption of healthy dietary habits has been shown to be beneficial to early cognitive development [38]. In the Ubicomp community, various applications designed to promote healthy behaviors, including diet, have been proposed and studied [28, 14, 27, 15].

Among those applications, the studies and products that target healthy meal recommendations have attracted much attention in academia [46] and industry [3] alike. Fundamentally, the goal of these systems is to suggest food alternatives that cater to individuals’ health goals and help users develop healthy eating behavior by following the recommendations [1].

Akin to most recommender systems, learning users’ preferences is a necessary step in recommending healthy meals that users are more likely to find desirable [1]. However, the current food preference elicitation approaches, including 1) on-boarding surveys and 2) food journaling, still suffer from many limitations. Specifically, a typical on-boarding survey asks a number of multi-choice questions about general food preferences. For example, PlateJoy [3], a daily meal planner app, elicits preferences for healthy goals and dietary restrictions with the following questions:

(1) How do you prefer to eat? No restrictions, dairy free, gluten free, kid friendly, pescatarian, paleo, vegetarian...

(2) Are there any ingredients you prefer to avoid? avocado, eggplant, eggs, seafood, shellfish, lamb, peanuts, tofu....

While the answers to these questions can certainly be used to create a rough dietary plan and avoid clearly unacceptable choices, they do not generate meal recommendations that cater to each person’s food preferences, and thus they suffer from lower recommendation acceptance rates, which is shown in our user study.

Food journaling is another popular approach for food preference elicitation. For example, Nutrino [2], a personal meal recommender, asks users to log their daily food consumptions and learn users’ fine-grained food preferences. With the proliferation of ubiquitous computing devices and their
increasing image-capture capability, food journaling is indeed a promising approach for fine-grained food preference learning. However, as a typical downside of systems relying on user-generated data, food journaling suffers from the cold-start problem, where recommendations cannot be made or are subject to low accuracy when the user has not yet generated a sufficient amount of data. For example, a previous study showed that an active food-journaling user makes about 3.5 entries per day [16]. It would take a non-trivial amount of time for the system to acquire sufficient amount of data to make recommendations, and the collected samples may subject to sampling biases as well [16]. Moreover, the habit of photo food journaling is difficult to adopt and adhere to [16].

In this work, we develop Yum-me, a meal recommender that learns users’ preferences based on general food images and then projects them into a healthy-food domain. We build upon a previously developed visual online learning framework, PlateClick [47], which learned users’ fine-grained food preferences with simple pairwise comparisons between food images. PlateClick did not address the challenge of using food preferences learned in the context of “general food” and applying it to recommendations of healthy meals, where users’ health goals and specific dietary restrictions need to be taken into account. Specifically, Yum-me learns users’ fine-grained food preferences through PlateClick’s adaptive visual interface and then attempts to generate meal recommendations that cater to the user’s health goals, food restrictions, as well as personal appetite for food. We present the design and implementation of Yum-me that can be used by people who have food restrictions, such as vegetarian, vegan, kosher, or halal, and have health goals for adjustment of calorie, fat, and protein intake.

For such a visual-based learning framework to work, one of the most critical components is a robust food image analysis model. Toward that end, as an additional contribution of this work we present a novel, unified food image analysis model called FoodDist. Based on deep convolutional networks and multi-task learning [31], FoodDist is the best-of-its-kind Euclidean distance embedding for food images, in which similar food items have smaller distances while dissimilar food items have larger distances. FoodDist allows Yum-me to learn users’ fine-grained food preferences accurately via similarity assessments on food images. Besides preference learning, FoodDist can be applied to other food-image-related tasks in ubiquitous computing, such as food image detection, classification, retrieval, and clustering. We benchmark FoodDist with the Food-101 dataset [10], the largest dataset for food images. The results suggest the superior performance of FoodDist over prior approaches (including the one proposed in the PlateClick paper) [47, 33, 10]. FoodDist is currently powering the backend of Yum-me and will be made available on Github.

We evaluate the desirability of Yum-me recommendations through a 60-person user study, where each user rates the healthy meal recommendations made by Yum-me relative to those made using a traditional survey-based approach. The study results show that, compared to the traditional survey based recommender, our system significantly improves the acceptance rate of the recommended healthy meals by 42.63%.

Note that Yum-me is not intended to be a replacement for general food surveys or food journaling. Instead, we argue that Yum-me is complementary to the existing systems, and is particularly beneficial in the cold-start phase. As an overview, we summarize our contributions to the Ubicomp community as follow:

- We present Yum-me, a healthy-meal recommender system that leverages users’ fine-grained food preferences to generate personalized recommendations that are superior in the context of their health goals, while being closer to their preferences than using traditional techniques.

- We develop FoodDist, a state-of-the-art unified food image analysis model that significantly outperforms all the existing models and can be applied to other ubiquitous dietary applications.

- We evaluate the system in a 60-person user study and demonstrate its capability in improving the desirability of recommended healthy options.

This paper presents an example of eliciting fine-grained preferences for recommender systems. The applicability of this personalization paradigm is not limited to food, but may also...
be applicable to general consumer products, content, entertainment, and IoT.

RELATED WORK

Healthy meals recommendation system

Traditional food and recipe recommendation systems learn users’ dietary preferences from users’ online activities, including ratings [21, 22], past recipe choices [41, 23], browsing history [44, 46, 2], etc. For example, Svensson et al. [41] built a social navigation system that recommends recipes based on the recipe choices made by different users. Van Pinxteren et al. [46] proposed to learn a recipe similarity measure from crowd card-sorting and make recommendations based on the self-reported meals. Beyond using users’ online activities, food logging and journaling learn users’ real food consumption history and require users’ active involvement, such as taking food images [16] or writing down ingredients and meta-information [46].

The above systems, while able to learn users’ detailed food preference, share a common limitation, that is, they need to wait until a user generates enough data before their recommendations can be effective for this user (i.e., the cold-start problem). Therefore, most commercial applications, for example, Zipongo [6] and Shopwell [4] adopt onboarding surveys to more quickly elicit users’ general food preferences. For instance, Zipongo’s questionnaires [6] ask users about their nutrient intake, lifestyle, habits, and food preferences, and then make day-to-day and week-to-week healthy meals recommendations; ShopWell’s survey [4] are designed to avoid certain food allergens, e.g., gluten, fish, corn, or poultry, and find meals that match to particular lifestyles, e.g., healthy pregnancy or athletic training.

Yum-me fills a vacuum that the prior approaches were not able to achieve, namely a rapid elicitation of users’ fine-grained food preferences for immediate healthy meal recommendations. Based on PlateClick’s online learning framework [47], Yum-me infers users’ preferences for each single food item among a large food dataset, and projects these preferences for general food items into the domain that meets each individual user’s health goals. We see Yum-me as a complement to the existing food preference elicitation approaches that further filters the food items selected by a traditional onboarding survey based on users’ fine-grained taste for food, and allows a system to serve tailored recommendations upon the first use of the system.

Ubiquitous food image analysis

The tasks of analyzing food images are very important in many ubiquitous dietary applications that actively or passively collect food images from mobile [16] and wearable [7, 42, 34] devices. The estimation of food intake and its nutritional information is helpful to our health [35] as it provides detailed records of our dietary history. Previous work mainly conducted the analysis by leveraging the crowd [35, 43] and computer vision algorithms [10, 33].

Noronha et al. [35] crowdsourced nutritional analysis of food images by leveraging the wisdom of untrained crowds. The study demonstrated the possibility of estimating a meal’s calories, fat, carbohydrates, and protein by aggregating the opinions from a large number of people; Turner-McGrievy et al. [43] elicit the crowd to rank the healthiness of several food items and validate the results against the ground truth provided by trained observers. Although this approach has been justified to be accurate, it inherently requires human resources that restrict it from scaling to large number of users and providing real time feedback.

To overcome the limitations of crowds and automate the analysis process, in the computer vision and Ubicomp communities, there are numerous papers discussing algorithms for food image analysis, including classification [10, 33, 29, 8], retrieval [30], and nutrient estimation [33, 40, 13, 24]. Most of the previous work [10] leveraged hand-crafted image features. However, traditional approaches were only demonstrated in special contexts, such as a specific restaurant [8] or particular type of cuisine [29] and the performances of the models might degrade when they are applied to food images in the wild.

In this paper, we designed FoodDist using deep convolutional neural network based multitask learning [12], which has been shown to be successful in improving model generalization power and performance in several applications [49, 17]. The main challenge of multitask learning is to design appropriate network structures and sharing mechanisms across tasks. With our proposed network structure, we show that FoodDist achieves superior performance when applied to the largest available real-world food image dataset [10], and when compared to prior approaches.

PERSONALIZED HEALTHY MEAL RECOMMENDATIONS

Our personalized healthy-meal recommendation system, Yumme, operates over a given inventory of food items and suggests the items that will appeal to the users’ palate and meet their health goals and dietary restrictions. A high-level overview of Yum-me’s recommendation process is shown in Fig. 1 and described as follows:

- First, users answer a simple survey to specify their dietary restrictions and health goals. This information is used by Yum-me to filter food items and create an initial set of recommendation candidates.
- Users then use PlateClick’s visual interface [47] to express their fine-grained food preferences through simple comparisons of food items. The learned preferences are used to further re-rank the recommendations and present to the users.

We describe five major components that enable the above recommendation process: 1) large-scale food datasets for different dietary restrictions, 2) the user survey, 3) food suitability ranking, 4) PlateClick visual interface, and 5) fine-grained preference re-ranking.

Large scale food database

To account for the dietary restrictions in many cultures and religions, or people’s personal choices, we prepare a separate food dataset for each of the following dietary restrictions:
No restrictions, Vegetarian, Vegan, Kosher, Halal

For each diet type, we pulled over 10,000 main dish recipes along with their images and metadata (ingredients, nutrients, tastes, etc.) from the Yummly API [5]. The total number of recipes is around 50,000. In order to customize food recommendations for people with specific dietary restrictions, e.g., vegetarian and vegan, we filter recipes by setting the allowedDiet parameter in the search API. For kosher or halal, we explicitly rule out certain ingredients by setting excludedIngredient parameter. The lists of excluded ingredients are shown as below:

- **Kosher**: pork, rabbit, horse meat, bear, shellfish, shark, eel, octopus, octopuses, moreton bay bugs, frog.
- **Halal**: pork, blood sausage, blood, blood pudding, alcohol, grain alcohol, pure grain alcohol, ethyl alcohol.

One challenge in using a public food image API is that many recipes returned by the API contain non-food images and incomplete nutritional information. Therefore, we further filter the items with the following criteria: the recipe should have 1) nutritional information of calories, protein and fat, and 2) at least one food image. In order to automate this process, we build a binary classifier based on a deep convolutional neural network to filter out non-food images. As suggested by [33], we treat the whole training set of Food-101 dataset [10] as one generic food category and sampled the same number of images (75,750) from the ImageNet dataset [18] as our non-food category. We took the pretrained VGG CNN model [39] and replaced the final 1000 dimensional softmax with a single logistic node. For the validation, we use the Food-101 testing dataset along with the same number of images sampled from ImageNet (25,250). We trained the binary classifier using the Caffe framework [26] and it reached 98.7% validation accuracy. We applied the criteria to all the datasets and the final statistics are shown in Table 1.

![Fig. 2 shows the visualizations of the collected datasets.](image)

For each of the recipe images, we embed it into a 1000-dimensional feature space using FoodDist (described later in FoodDist section) and then project all the images onto a 2-D plane using t-Distributed Stochastic Neighbor Embedding (t-SNE) [45]. For visibility, we further divide the 2-D plane into several blocks; from each of which, we sample a representative food image residing in that block to present in the figure. Fig. 2 demonstrates the large diversity and coverage of the collected datasets. Also, the embedding results clearly demonstrate the effectiveness of FoodDist in grouping similar food items together while pushing dissimilar items away. This is important to the performance of Yum-me, as discussed in User Study section.

### User survey

The user survey is designed to elicit user’s high-level dietary restrictions and health goals. Users can specify their dietary restrictions among the five categories mentioned-above and indicate their health goals in terms of the desired amount of **calories**, **protein** and **fat**. We choose these nutrients for their high relevance to many common health goals, such as weight control [19], sports performance [11], etc. We provide three options for each of these nutrients, including **reduce**, **maintain**, and **increase**. The user’s diet type is used to select the appropriate food dataset, and the food items in the dataset are further ranked by their suitability to users’ health goals based on the nutritional facts.

#### Food suitability ranking

To measure the suitability of food items given users’ health goals, we rank the recipes in terms of different nutrients in both ascending and descending order, such that each recipe is associated with six ranking values, i.e., $r_{calories,a}$, $r_{calories,d}$, $r_{protein,a}$, $r_{protein,d}$, $r_{fat,a}$ and $r_{fat,d}$, where $a$ and $d$ stand for ascending and descending respectively. The final suitability value for each recipe given the health goal is calculated as follows:

$$ u = \sum_{n \in U} a_{n,a} r_{n,a} + \sum_{n \in U} a_{n,d} r_{n,d} $$

where $U = \{\text{calories, protein, fat}\}$. The indicator coefficient $a_{n,a} = 1 \iff \text{nutrient n is rated as reduce and } a_{n,d} = 1 \iff \text{nutrient n is rated as increase}$. Otherwise $a_{n,a} = 0$ and $a_{n,d} = 0$. Finally, given a user’s responses to the survey, we rank the suitability of all the recipes in the corresponding database and select top-$M$ items (around top 10%) as the candidate pool of healthy meals for this user. In our initial prototype, we set $M = 500$.

#### PlateClick: fine-grained preference elicitation

Based on the food suitability ranking, a candidate pool of healthy meals is created. However, not all the meals in this candidate pool will suit the user’s palate. Therefore, PlateClick is used to further identify recipes that cater to the user’s taste through eliciting their fine-grained food preferences. PlateClick is a visual online learning framework proposed in [47]. It learns users’ fine-grained food preferences by presenting users with food images and ask them to choose ones that look delicious. In the following, we provide a brief introduction to PlateClick’s online learning algorithm; for details, see [47].

Formally, the food preference learning task can be defined as: given a large target set of food items $\mathbb{S}$, we represent user’s preferences as a distribution over all the possible food items, i.e. $p = [p_1, ..., p_|\mathbb{S}|], \sum_i p_i = 1$, where each element $p_i$ denotes

| Database       | Original size | Final size |
|----------------|---------------|------------|
| No restriction | 9405          | 7938       |
| Vegetarian     | 10000         | 6713       |
| Vegan          | 9638          | 6013       |
| Kosher         | 10000         | 4825       |
| Halal          | 10000         | 5002       |

Table 1. Sizes of databases that catered to different diet types. Unit: number of unique recipes.
the user’s favorable scale for item $i$. Since the number of items, $|\mathcal{S}|$, is usually quite large and intractable to elicit each $p_i$ directly from the user $^3$, the approach PlateClick takes is to adaptively choose a much smaller subset $\mathcal{V}$ to present to the user, and propagate the users’ preferences for those items to the rest items based on their visual similarity. Specifically, as Fig.1 shows, PlateClick’s adaptive preference elicitation process can be divided into two phases:

**Phase I:** In each of the first 2 iterations, we present ten food images and ask users to tap all the items that look delicious to them.

**Phase II:** In each of the subsequent 13 iterations, we present a pair of food images and ask users to either compare the food pair and tap on the one that looks delicious to them or tap on “Yuck” if neither of the items appeal to their taste.

The prior study showed that $|\mathcal{V}| \leq 46 \ll |\mathcal{S}|$ is sufficient to learn users’ preferences if images $\mathcal{V}$ are chosen properly. PlateClick adaptively chooses image sets for each iteration based on an exploration-exploitation strategy such that the images can cover a diverse range of food items and efficiently explore users’ preferences. After each iteration $t$, we update the preference distribution $\mathbf{p}^{t-1}$ to $\mathbf{p}^t$ using label propagation based on user’s selections, where $\mathbf{p}^t$ denotes the updated preference distribution at iteration $t$. The probability that the preference for food $i$ will be propagated to food $j$ is determined by the similarity between them, denoted by $w_{ij}$, defined as follows:

$$w_{ij} = \begin{cases} e^{-\frac{1}{2\alpha^2} \| f(x_i) - f(x_j) \|^2} & \| f(x_i) - f(x_j) \| \leq \delta \\ 0 & \| f(x_i) - f(x_j) \| > \delta \end{cases}$$

where $\alpha^2 = \frac{1}{|\mathcal{S}|^2} \sum_{i,j \in \mathcal{S}} \| f(x_i) - f(x_j) \|^2$. $f(x_i)$ and $f(x_j)$ are feature vectors of food images $x_i$ and $x_j$ extracted by a feature extractor, denoted by $f$, and $\| f(x_i) - f(x_j) \|$ represents the euclidean distance between item $i$ and item $j$ in the embedding.

$^3$The target set is often the whole food database that different applications use. For example, the size of Yummly database can be up to 1-million [5].

We use FoodDist as the feature extractor such that if the food items are similar to each other (i.e. $\| f(x_i) - f(x_j) \|$ is close to 0), then $w_{ij}$ is approximately 1. Otherwise, with the increasing of the pairwise distance, the value of $w_{ij}$ smoothly goes down and finally reaches 0 when $\| f(x_i) - f(x_j) \|$ is larger than a certain threshold $\delta$. More details about FoodDist appear in FoodDist section.

**Preference re-ranking**

Through PlateClick, we learn users’ general food preferences. In order to project this information into the domain of healthy meal recommendations, we further re-rank the $M$ candidate healthy food items with the users’ preferences (i.e. $p_i$ for $i \in M$), and pick the top $N$ items to present to the user. Note that, the general food preferences learned by PlateClick can be used not only for healthy meal recommendations but also for other dietary applications through a similar projection scheme. Also, in order to prevent repeating items between preferences learning and recommendation, in the recommendation phase, we remove those food images that are already used in PlateClick.

Fundamentally, for Yummly to work, a robust feature extractor $f$ for food is essential because both the accuracy of the inferred food preferences $\mathbf{p} = [p_1, ..., p_{|\mathcal{S}|}]$ and the effectiveness of PlateClick’s online learning algorithm all depend on a feature extractor $f$ that can extract a robust feature vector $f(x_i)$ for every image $i$ such that the food images’ euclidean distances reflect their similarity. In the next section, we present such a model: FoodDist and discuss its broader applicability to food image analysis tasks in dietary applications.

**FOODDIST**

Formally, the goal of FoodDist is to learn a feature extractor (embedding) $f$ such that given an image $x$, $f(x)$ will project it to an $N$ dimensional feature vector and the euclidean distances between vectors reflect the similarities between food images, as Fig.3 shows. Formally speaking, if image $x_1$ is more similar to image $x_2$ than image $x_3$, $\| f(x_1) - f(x_2) \| < \| f(x_1) - f(x_3) \|$.

We build FoodDist based on recent advances in deep Convolutional Neural Networks (CNN), which provides a powerful
framework for automatic feature learning. Traditional feature representations for images are mostly hand-crafted, and were used with feature descriptors, such as SIFT (Scale Invariant Feature Transform)[32] etc., which are invariant to object scale and illumination and improve the generalizability of the trained model. However, when image characteristics are diverse the one-size-fits-all feature extractor performs poorly. In contrast, deep learning adapts the features to particular image characteristics and extracts features that are most discriminative in the given task [37].

As we present below, a feature extractor for food images can be learned through classification and metric learning, or through multitask learning, which concurrently performs the former two tasks. We demonstrate that the proposed multitask learning approach enjoys the benefits of both classification and metric learning, and achieves the best performance.

Learning with classification
One common way to learn a feature extractor for labeled data is to train a neural network that performs classification (i.e. mapping input to labels), and takes the output of a hidden layer as the feature representations; specifically, using a feedforward deep convolutional neural network with n-layers (as the upper half of the Fig. 4 shows):

$$f(x) = g_n(g_{n-1}(\ldots g_1(x)\ldots))$$  \hspace{1cm} (3)$$

where $g_i(\cdot)$ represents the computation of $i$-th layer (e.g. convolution, pooling, fully-connected, etc.), and $F(x)$ is the output class label. The difference of the output class label to the ground truth (i.e. the error) is back-propagated throughout the whole network from layer $n$ to the layer 1. We can take the output of the layer $n-1$ as the feature representation of $x$, which is equivalent to having a feature extractor $f$ as:

$$f(x) = g_{n-1}(\ldots g_1(x)\ldots)$$  \hspace{1cm} (4)$$

Usually, the last few layers will be fully-connected layers, and the last layer $g_n(\cdot)$ is roughly equivalent to a linear classifier that is built on the features $f(x)$ [25]. Therefore, $f(x)$ is discriminative in separating instances under different categorical labels, and the euclidean distances between normalized feature vectors can reflect the similarities between images.

Metric Learning
Different from the classification approach, where the feature extractor is a by-product, metric learning proposes to learn the distance embedding directly from the paired inputs of similar and dissimilar examples. Prior work [47] used a Siamese network to learn a feature extractor for food images. The structure of a siamese network resembles that in Fig. 4 but without Class label, Fully connected, 101 and Softmax layers. The inputs of the siamese network are a pair of food images $x_1, x_2$. Each image goes through an sharing weight CNN and the output of each network is regarded as the feature representation, i.e. $f(x_1)$ and $f(x_2)$, respectively. Our goal is for $f(x_1)$ and $f(x_2)$ to have a small distance value (close to 0) if $x_1$ and $x_2$ are similar food items; otherwise, they should have a larger distance value. The value of contrastive loss is then back-propagated to optimize the siamese network:

$$L(x_1, x_2, l) = \frac{1}{2}D^2 + \frac{1}{2}(1-l)\max(0, m-D)^2$$ \hspace{1cm} (5)$$

where similarity label $l \in \{0, 1\}$ indicates whether the input pair of food items $x_1, x_2$ are similar or not ($l = 1$ for similar, $l = 0$ for dissimilar), $m > 0$ is the margin for dissimilar items and $D$ is the Euclidean distance between $f(x_1)$ and $f(x_2)$ in embedding space. Minimizing the contrastive loss will pull similar pairs together and push dissimilar farther away (larger than a margin $m$) and it exactly matches the goal.

The major advantage of metric learning is that the network will be directly optimized for our final goal, i.e. a robust distance measure between images. However, as shown in the model benchmarks, using the pairwise information alone does not improve the embedding performance as the process of sampling pairs loses the label information, which is arguably more discriminative than (dis)similar pairs.

Multitask Learning: concurrently optimize both tasks
Both methods above have their pros and cons. Learning with classification leverages the label information, but the network is not directly optimized to our goal. As a result, although the feature vectors are learned to be separable in the linear space, the intra- and inter- categorical distances might still be unbalanced. On the other hand, metric learning is explicitly optimized for our final objective by pushing the distances between dissimilar food items larger than a margin $m$. Nevertheless, sampling the similar or dissimilar pairs loses much label information. For example, given a pair of items with different labels, we only consider the dissimilarity between the two categories they belong to, but overlook the fact that each
item is also different from the rest \( n - 2 \) categories, where \( n \) is total number of categories.

In order to leverage the benefits of both tasks, we propose a multitask learning design [25] for FoodDist. The idea of multitask learning is to share part of the model across tasks so as to improve the generalization ability of the learned model [25]. In our case, as Fig.4 shows, we share the parameters between the classification network and siamese network, and optimize them simultaneously. We use the base structure of the siamese network and share the upper CNN with a classification network where the output of the CNN is fed into a cascade of fully connected layer and softmax loss layer. The final loss of the whole network is the weighted sum of the softmax loss \( L_{\text{softmax}} \) and contrastive loss \( L_{\text{contrastive}} \):

\[
L = \omega L_{\text{softmax}} + (1 - \omega) L_{\text{contrastive}} \tag{6}
\]

Our benchmark results suggest that the feature extractor built with multitask learning achieves the best of both worlds: i.e. it achieves the best performance for both classification and euclidean distance based retrieval tasks. In the following, we present the benchmark methodology and results.

**Model details and benchmark setup**

We train all the models using Food-101 training dataset, which contains 75,750 food images from 101 food categories (750 instances for each category) [10]. To the best of our knowledge, Food-101 is the largest and most challenging publicly available dataset for food images. We implement models using Caffe[26] and experiment two CNN architectures in our framework: AlexNet[31], which won the first place at ILSVRC2012 challenge, and VGG[39], which is the state-of-the-art CNN model. The inputs to the networks are image crops of sizes \( 224 \times 224 \) (VGG) or \( 227 \times 227 \) (AlexNet). They are randomly sampled from an pixelwise-mean-subtracted image or its horizontal flip. In our benchmark, we train four different feature extractors: AlexNet+Learning with classification (AlexNet+CL), AlexNet+Multitask learning (AlexNet+MT), VGG+Learning with classification (VGG+CL) and VGG+Multitask learning (VGG+ML, FoodDist). For the multitask learning framework, we sample the similar and dissimilar image pairs with 1:10 ratio from the Food-101 dataset based on the categorical labels to be consistent with the previous work[47]. The models are fine-tuned based on the networks pre-trained with the ImageNet data. We use Stochastic Gradient Decent with a mini-batch size of 64, and each network is trained for \( 10 \times 10^4 \) iterations. The initial learning rate is set to 0.001 and we use a weight decay of 0.0005 and momentum of 0.9.

We compare the performance of four feature extractors, including FoodDist, with the state-of-the-art food image analysis models using Food-101 testing dataset, which contains 25,250 food images from 101 food categories (250 instances for each category). The performance for classification and retrieval tasks are evaluated as follows:

- **Classification:** We test the performance of using learned image features for classification. For the classification deep neural network in each of the models above, we adopt the standard 10-crop testing, i.e. the network makes a prediction by extracting ten patches (the four corner patches and the center patch in the original images and their horizontal reflections), and averaging the predictions at the softmax layer. The metrics used in this paper are Top-1 accuracy and Top-5 accuracy.

- **Retrieval:** We use a retrieval task to evaluate the quality of the euclidean distances between extracted features. Ideally, the distances should be smaller for similar image pairs and
larger for dissimilar pairs. Therefore, as suggested by previous work [47, 48], we check the nearest k-neighbors of each test image, for $k = 1, 2, ..., N$, where $N = 25250$ is the size of the testing dataset, and calculate the Precision and Recall values for each $k$. We use mean Average Precision (mAP) as the evaluation metric to compare the performance. For every method, the Precision/Recall values are averaged over all the images in the testing set.

**Performance**

The classification and retrieval performance of all models are summarized in Table 2 and Table 3 respectively. FoodDist performs the best among four models and is significantly better than the state-of-the-art approaches in both tasks. For the classification task, the classifier built on FoodDist features achieves 83.09% Top-1 accuracy, which significantly outperforms the original RFDC [10] model and the proprietary GoogLeNet model [33]; For the retrieval task, FoodDist doubles the mAP value reported by previous work [47] that only used the AlexNet and siamese network architecture. The benchmark results demonstrate that FoodDist features possess high generalization ability and the euclidean distances between feature vectors reflect the similarities between food images with great fidelity. In addition, as we can observe from both tables, the multitask learning based approach always performs better than learning with classification for both tasks no matter which CNN is used. This further justifies the proposed multitask learning approach and its advantage of incorporating both label and pairwise distance information that makes the learned features more generalizable and meaningful in the euclidean distance embedding.

| Method         | Top-1 ACC (%) | Top-5 ACC(%) |
|----------------|---------------|--------------|
| RFDC*[10]      | 50.76%        | --           |
| GoogleLeNet*[33]| 79%           | --           |
| AlexNet+CL     | 67.63%        | 89.02%       |
| AlexNet+MT     | 70.50%        | 90.36%       |
| VGG+CL         | 82.48%        | 95.70%       |
| VGG+MT (FoodDist) | 83.09%     | 95.82%       |

Table 2. Model performance of classification task. * represents state-of-the-art approach and bold text indicates the method with the best performance.

| Method         | mean Average Precision (mAP) |
|----------------|-----------------------------|
| Food-CNN*[47]  | 0.3084                      |
| AlexNet+CL     | 0.3751                      |
| AlexNet+MT     | 0.4063                      |
| VGG+CL         | 0.6417                      |
| VGG+MT (FoodDist) | 0.6670                     |

Table 3. Model performance of retrieval task. * represents state-of-the-art approach and bold text indicates the method with the best performance. (Note: The mAP value that we report for Food-CNN is higher because we use pixel-wise mean subtraction while the original paper only use per-channel mean subtraction.)

FoodDist for a wide range of food image analysis tasks

Essentially, FoodDist provides a unified model to extract features from food images so that they are discriminative in the classification and clustering tasks, and its pairwise euclidean distances are meaningful in reflecting similarities. The model is rather efficient (< 0.5s/f on 8-core commodity processors) and can be ported to mobile devices with the publicly-available caffe-android-lib framework.

In addition to enabling Yumme, we plan to release FoodDist model and API to the community that can be used to fuel other nutritional applications. For the sake of space, we only briefly discuss two sample use cases below:

- **Food/Meal recognition**: Given a set of labels, e.g. food categories, cuisines, and restaurants etc., the task of food and meal recognition could be approached by first extracting food image features from FoodDist and then train a linear classifier, e.g. Logistic regression, SVM etc., to classify the food images that are beyond the categories given in the Food-101 dataset.

- **Nutritional fact estimation**: With the emergence of large-scale food item or recipe databases, such as Yummly, the problem of nutritional facts estimation might be converted to a simple nearest-neighbor retrieval task: i.e. given a query image, we find its closest neighbor in the FoodDist based on their euclidean distance, and use that neighbor’s nutritional information to estimate the nutrition facts of the query image.

**USER STUDY**

We conducted a user study to validate the efficacy of Yumme recommendations. We recruited 60 participants through the university mailing list, Facebook, and Twitter. The goal of the user study was to compare Yumme recommendations with a widely-used baseline approach, i.e. a traditional food preference survey. We used a within-subjects study design in which each participant expressed their opinions regarding the meals recommended by both of the recommenders, and the effectiveness of the systems were compared on a per-user basis.

**Study Design**

We created a traditional recommendation system by randomly picking $N$ out of $M$ meals in the candidate pool to recommend to the users. The values of $N$ and $M$ are controlled such that $N = 10, M = 500$ for both Yumme and the traditional
baseline. The user study consists of three phases, as Fig.5 shows: (1) Each participant was asked to indicate their diet type and health goals through our basic user survey. (2) Each participant was then asked to use PlateClick. (3) 20 meal recommendations were arranged in a random order and presented to the participant at the same time, where 10 of them are made by Yum-me, and the other 10 are generated by the baseline. The participant was asked to express their opinion by dragging each of the 20 meals into either the Yummy or the No way bucket. To overcome the fact that humans tend to balance the buckets if their previous choices were shown, the food item disappeared after the user dragged it into a bucket. In this way, users were not reminded of how many meals they had put into each bucket.

The user study systems were implemented as web services and participants accessed the study from desktop or mobile browsers. We chose a web service for its wide accessibility to the population, but we could easily fit Yum-me into other ubiquitous devices, as mentioned earlier.

Participants
The most common dietary choice among our 60 participants was No restrictions (48), followed by Vegetarian (9), Halal (2) and Kosher (1). No participants chose Vegan. Participant preferences in terms of nutrients are summarized in Table 4. For Calories and Fat, the top two goals were Reduce and Maintain. For Protein, participants tended to choose either Increase or Maintain. For health goals, the top four participant choices were Maintain calories-Maintain protein-Maintain fat (20); Reduce calories-Maintain protein-Reduce fat (10), Reduce calories-Maintain protein-Maintain fat (10) and Reduce calories-Increase protein-Reduce fat (5). The statistics match well with the common health goals among the general population, i.e. people who plan to control weight and improve sports performance tend to reduce the intake calories and fat, and increase the amount of protein.

Quantitative analysis
We calculated the participant acceptance rate of meal recommendations as: \[
\text{Acceptance rate} = \frac{\# \text{Meals in Yummy bucket}}{\# \text{Recommended meals}}
\]

The cumulative distribution of the acceptance rate differences between two recommender systems is shown in Fig.6 and the average values per approach are presented in Table 5. The results clearly demonstrate that Yum-me significantly improves the quality of the presented food items. The per-user difference between the two approaches was normally distributed and a paired Student’s t-test indicated a significant difference between the two methods (\( p < 0.0001 \)).

To further quantify the improvement provided by the Yum-me system, we calculated the difference between the acceptance rates of the two systems, i.e. \( \text{difference} = \text{Yum-me acceptance rate} - \text{baseline acceptance rate} \). The distribution and average values of the differences are presented in Fig.7 and Table 5 respectively. It is noteworthy that Yum-me

| Nutrient | Reduce | Maintain | Increase |
|----------|--------|----------|----------|
| Calories | 30     | 28       | 2        |
| Protein  | 1      | 44       | 15       |
| Fat      | 23     | 36       | 1        |

Table 4. Statistics of health goals among 60 participants. Unit: number of participants.

Figure 7. Distribution of the acceptance rate differences between two recommender systems.

5 A Shapiro Wilk W test was not significant (\( p = 0.12 \)), which justifies that the difference is normally distributed.

6 We also performed a non-parametric Wilcoxon signed-rank test and found a comparable result.
outperformed the baseline by 42.63% in terms of the number of preferred recommendations, which demonstrates its utility over the traditional meal recommendation approach.

| Metric           | Mean   | SEM   |
|------------------|--------|-------|
| Yum-me avg. acc. | 0.7250 | 0.0299|
| Baseline avg. acc.| 0.5083 | 0.0341|
| Avg. acc. difference | 0.2167 | 0.0297|

Table 5. Average acceptance rates and average difference between two systems. Paired t-test p-value: $8.58 \times 10^{-10}$; (acc.: acceptance rate)

**Qualitative analysis**

To qualitatively understand the personalization mechanism of Yum-me, we randomly pick 3 participants with no dietary restrictions and with the health goal of reducing calories. For each such user, we select top-20 general food items the user likes most (inferred by PlateClick). These food items played important roles in selecting the healthy meals to recommend to the user. To visualize this relationship, among these top-20 items, we further select two food items that are most similar to the healthy items Yum-me recommended to the users and present three such examples in Fig.8. Intuitively, our system is able to recommend healthy food items that are visually similar to the food items a user likes, but the recommended items are of lower calories due to the use of healthier ingredients or different cooking styles. These examples showcase how Yum-me can project users’ general food preferences to the domain of the healthy options, and find the ones that can most appeal to users.

**Error analysis**

Through a closer examination of the cases where our system performed relatively poorly, we observed a negative corre-

![Figure 8. Qualitative analysis of personalized healthy meal recommendations. Images on the left half are sampled from users’ top-20 favorite meals learned from Yum-me; Images on the right half are the meals presented to the user. The number under each food image represents the amount of calories for the dish, unit: kcal/serving.](image)

![Figure 9. Entropy of preference distributions in different iterations of PlateClick. (Data from 48 users with no dietary restrictions)](image)

lation between the entropy of the learned preference distribution $p$ and the improvement of Yum-me over the baseline ($r = -0.32, p = 0.026$). This correlation suggests that when user’s preference distributions are more concentrated, the recommended meals tend to perform better. This is not too surprising because the entropy of the preference distribution roughly reflects the degree of confidence the system has in the users’ preferences, where the confidence is higher if the entropy is lower and vice versa. In Fig.9, we show the evolution of the entropy value as the users are making more comparisons through PlateClick. The results demonstrate that the system becomes more confident about user’s preferences as users provide more feedback.

**CONCLUSION AND FUTURE WORK**

In this paper, we propose Yum-me, a novel healthy meal recommender that makes health meal recommendations catering to users’ fine-grained food preferences. We further present FoodDist, a best-of-its-kind unified food image analysis model. The user study and benchmarking results demonstrate the effectiveness of Yum-me and superior performance of FoodDist model. Looking forward, we envision that the idea of using visual similarity for preference elicitation can power a wider range of applications in the Ubicomp community. (1) User-centric dietary profile: the fine-grained food preference learned by Yum-me can be seen as a general dietary profile of each user and be projected to other domains to enable more dietary applications, such as suggesting proper meal plans for diabetes patients. Moreover, a personal dietary API can be built on top of this profile to enable the sharing and improvements across multiple dietary applications. (2) Food image analysis API for deeper content understanding: With the release of the FoodDist model and API, many dietary applications, in particular the ones that capture a large number of food images, might benefit from a deeper understanding of their image contents. For instance, food journaling applications could benefit from the automatic analysis of food images to summarize the day-to-day food intake or trigger timely reminders and suggestions when needed. (3) Fine-grained preference elicitation leveraging visual interfaces. The idea of eliciting users’ fine-grained preference via visual interfaces is also applicable to other domains in Ubicomp. The key insight here is that visual contents capture many subtle

$^7$Entropy of preference distribution: $H(p) = -\sum p_i \log p_i$
variations among objects that text or categorical data cannot capture; and the learned representations can be used as an effective medium to enable fine-grained preferences learning. For instance, the IoT, wearable, and mobile systems for entertainments, consumer products, and general content deliveries can leverage such an adaptive visual interface to design an onboarding process that learns users’ preferences in a much shorter time and potentially more pleasantly than the traditional approaches.

REFERENCES

1. 2015. Personalizing Food Recommendations with Data Science. http://blog.zipongo.com/blog/2015/8/11/personalizing-food-recommendations-with-data-science. (2015).
2. 2016. nutrino. http://nutrino.co/. (2016).
3. 2016. PlateJoy. https://www.platejoy.com/. (2016).
4. 2016. ShopWell. http://www.shopwell.com/. (2016).
5. 2016. Yummly. http://developer.yummly.com. (2016).
6. 2016. Zipongo. https://zipongo.com/. (2016).
7. Lenore Arab, Deborah Estrin, Donnie H Kim, Jeff Burke, and Jeff Goldman. 2011. Feasibility testing of an automated image-capture method to aid dietary recall. European journal of clinical nutrition 65, 10 (2011), 1156–1162.
8. Oscar Beijbom, Neel Joshi, Dan Morris, Scott Saponas, and Siddharth Khullar. 2015. Menu-match: restaurant-specific food logging from images. In Applications of Computer Vision (WACV), 2015 IEEE Winter Conference on. IEEE, 844–851.
9. Lisa M Bodnar and Katherine L Wisner. 2005. Nutrition and depression: implications for improving mental health among childbearing-aged women. Biological psychiatry 58, 9 (2005), 679–685.
10. Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. 2014. Food-101—mining discriminative components with random forests. In Computer Vision–ECCV 2014. Springer, 446–461.
11. JR Brotherhood. 1984. Nutrition and sports performance. Sports Medicine 1, 5 (1984), 350–389.
12. Rich Caruana. 1997. Multitask learning. Machine learning 28, 1 (1997), 41–75.
13. Junghoon Chae, Insoo Woo, SungYe Kim, Ross Maciejewski, Fengqing Zhu, Edward J Delp, Carol J Boushey, and David S Ebert. 2011. Volume estimation using food specific shape templates in mobile image-based dietary assessment. In IS&T/SPIE Electronic Imaging. International Society for Optics and Photonics, 78730K–78730K.
14. Kerry Shih-Ping Chang, Catalina M Danis, and Robert G Farrell. 2014. Lunch Line: using public displays and mobile devices to encourage healthy eating in an organization. In Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing. ACM, 823–834.
15. Sunny Consolvo, David W McDonald, Tammy Toscos, Mike Y Chen, Jon Froehlich, Beverly Harrison, Predrag Klasonja, Anthony LaMarka, Louis LeGrand, Ryan Libby, and others. 2008. Activity sensing in the wild: a field trial of ubifit garden. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 1797–1806.
16. Felicia Cordeiro, Elizabeth Bales, Erin Cherry, and James Fogarty. 2015. Rethinking the mobile food journal: Exploring opportunities for lightweight photo-based capture. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems. ACM, 3207–3216.
17. Jifeng Dai, Kaiming He, and Jian Sun. 2015. Instance-aware Semantic Segmentation via Multi-task Network Cascades. arXiv preprint arXiv:1512.04412 (2015).
18. Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. IEEE, 248–255.
19. Leonard H Epstein, Rena R Wing, Barbara C Penner, and Mary Jeanne Kress. 1985. Effect of diet and controlled exercise on weight loss in obese children. The Journal of Pediatrics 107, 3 (1985), 358–361.
20. Centers for Disease Control, Prevention (CDC, and others. 2011. School health guidelines to promote healthy eating and physical activity. MMWR. Recommendations and reports: Morbidity and mortality weekly report. Recommendations and reports/Centers for Disease Control 60, RR-5 (2011), 1.
21. Peter Forbes and Mu Zhu. 2011. Content-boosted matrix factorization for recommender systems: experiments with recipe recommendation. In Proceedings of the fifth ACM conference on Recommender systems. ACM, 261–264.
22. Jill Freyne and Shlomo Berkovsky. 2010. Intelligent food planning: personalized recipe recommendation. In Proceedings of the 15th international conference on Intelligent user interfaces. ACM, 321–324.
23. Gijs Geleijnse, Peggy Nachtigall, Pim van Kaam, and Luciënne Wijgergangs. 2011. A personalized recipe advice system to promote healthful choices. In Proceedings of the 16th international conference on Intelligent user interfaces. ACM, 437–438.
24. Ye He, Chang Xu, Neha Khanna, Carol J Boushey, and Edward J Delp. 2013. Food image analysis: Segmentation, identification and weight estimation. In Multimedia and Expo (ICME), 2013 IEEE International Conference on. IEEE, 1–6.
25. Yoshua Bengio Ian Goodfellow and Aaron Courville. 2016. Deep Learning. (2016). http://goodfeli.github.io/dlbook/ Book in preparation for MIT Press.

26. Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. 2014. Caffe: Convolutional architecture for fast feature embedding. In Proceedings of the ACM International Conference on Multimedia. ACM, 675–678.

27. Azusa Kadomura, Cheng-Yuan Li, Yen-Chang Chen, Hao-Hua Chu, Koji Tsukada, and Itiro Siio. 2013. Sensing fork and persuasive game for improving eating behavior. In Proceedings of the 2013 ACM conference on Pervasive and ubiquitous computing adjunct publication. ACM, 71–74.

28. Azusa Kadomura, Cheng-Yuan Li, Koji Tsukada, Hao-Hua Chu, and Itiro Siio. 2014. Persuasive technology to improve eating behavior using a sensor-embedded fork. In Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing. ACM, 319–329.

29. Yoshiyuki Kawano and Keiji Yanai. 2014. Food image recognition with deep convolutional features. In Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication. ACM, 589–593.

30. Keigo Kitamura, Toshihiko Yamasaki, and Kiyoharu Aizawa. 2009. FoodLog: capture, analysis and retrieval of personal food images via web. In Proceedings of the ACM multimedia 2009 workshop on Multimedia for cooking and eating activities. ACM, 23–30.

31. Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems. 1097–1105.

32. David G Lowe. 2004. Distinctive image features from scale-invariant keypoints. International journal of computer vision 60, 2 (2004), 91–110.

33. Austin Meyers, Nick Johnston, Vivek Rathod, Anoop Korattikara, Alex Gorban, Nathan Silberman, Sergio Guadarrama, George Papandreou, Jonathan Huang, and Kevin P Murphy. 2015. Im2Calories: towards an automated mobile vision food diary. In ICCV. 1233–1241.

34. Kher Hui Ng, Victoria Shipp, Richard Mortier, Steve Benford, Martin Plintham, and Tom Rodden. 2015. Understanding food consumption lifecycles using wearable cameras. Personal and Ubiquitous Computing 19, 7 (2015), 1183–1195.

35. Jon Noronha, Eric Hysen, Haoqi Zhang, and Krzysztof Z Gajos. 2011. Platemate: crowdsourcing nutritional analysis from food photographs. In UIST. ACM, 1–12.

36. Rachel Clare Povey and David Clark-Carter. 2007. Diabetes and Healthy Eating A Systematic Review of the Literature. The Diabetes Educator 33, 6 (2007), 931–959.

37. Ali Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. 2014. CNN features off-the-shelf: an astounding baseline for recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. 806–813.

38. Jonathan Shepherd, Angela Harden, Rebecca Rees, Ginny Brunton, Jo Garcia, Sandy Oliver, and Ann Oakley. 2006. Young people and healthy eating: a systematic review of research on barriers and facilitators. Health Education Research 21, 2 (2006), 239–257.

39. K. Simonyan and A. Zisserman. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. CoRR abs/1409.1556 (2014).

40. Kyoko Sudo, Kazuhiko Murasaki, Jun Shimamura, and Yukinobu Taniguchi. 2014. Estimating nutritional value from food images based on semantic segmentation. In Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication. ACM, 571–576.

41. Martin Svensson, Kristina Höök, and Rickard Cöster. 2005. Designing and evaluating kalas: A social navigation system for food recipes. ACM Transactions on Computer-Human Interaction (TOCHI) 12, 3 (2005), 374–400.

42. Edison Thomaz, Aman Parnami, Irfan Essa, and Gregory D Abowd. 2013. Feasibility of identifying eating moments from first-person images leveraging human computation. In Proceedings of the 4th International SenseCam & Pervasive Imaging Conference. ACM, 26–33.

43. Gabrielle M Turner-McGrievy, Elina E Helander, Kirsikka Kaipainen, Jose Maria Perez-Macias, and Ilkka Korhonen. 2015. The use of crowdsourcing for dietary self-monitoring: crowdsourced ratings of food pictures are comparable to ratings by trained observers. Journal of the American Medical Informatics Association 22, e1 (2015), e112–e119.

44. Mayumi Ueda, Syungo Asanuma, Yusuke Miyawaki, and Shinsuke Nakajima. 2014. Recipe recommendation method by considering the user's preference and ingredient quantity of target recipe. In Proceedings of the International MultiConference of Engineers and Computer Scientists, Vol. 1.

45. Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. Journal of Machine Learning Research 9, 2579-2605 (2008), 85.

46. Youri van Pinxteren, Gijs Geleijnse, and Paul Kamsteeg. 2011. Deriving a recipe similarity measure for recommending healthful meals. In Proceedings of the 16th international conference on Intelligent user interfaces. ACM, 105–114.
47. Longqi Yang, Yin Cui, Fan Zhang, John P Pollak, Serge Belongie, and Deborah Estrin. 2015a. PlateClick: Bootstrapping Food Preferences Through an Adaptive Visual Interface. In Proceedings of the 24th ACM International on Conference on Information and Knowledge Management. ACM, 183–192.

48. Longqi Yang, Cheng-Kang Hsieh, and Deborah Estrin. 2015b. Beyond Classification: Latent User Interests Profiling from Visual Contents Analysis. arXiv preprint arXiv:1512.06785 (2015).

49. Zhanpeng Zhang, Ping Luo, Chen Change Loy, and Xiaoou Tang. 2014. Facial landmark detection by deep multi-task learning. In Computer Vision–ECCV 2014. Springer, 94–108.