What's in a Name? First Names as Facial Attributes

Huizhong Chen
Stanford University
hchen2@stanford.edu

Andrew C. Gallagher
Cornell University
andrew.c.gallagher@cornell.edu

Bernd Girod
Stanford University
bgirod@stanford.edu

Abstract

This paper introduces a new idea in describing people using their first names, i.e., the name assigned at birth. We show that describing people in terms of similarity to a vector of possible first names is a powerful description of facial appearance that can be used for face naming and building facial attribute classifiers.

We build models for 100 common first names used in the United States and for each pair, construct a pairwise first-name classifier. These classifiers are built using training images downloaded from the internet, with no additional user interaction. This gives our approach important advantages in building practical systems that do not require additional human intervention for labeling. We use the scores from each pairwise name classifier as a set of facial attributes.

We show several surprising results. Our name attributes predict the correct first names of test faces at rates far greater than chance. The name attributes are applied to gender recognition and to age classification, outperforming state-of-the-art methods with all training images automatically gathered from the internet.

1. Introduction

Expectant parents spend a great deal of time selecting a first name for their child. To the parents, this choice may appear to be a selection from a near-infinite pool of possibilities. However, social context influences this decision, from the obvious factors (e.g., gender), to the less obvious ones (e.g., ethnicity, socio-economic background, popularity of names, names of relatives and friends). Consequently, first names are not distributed at random among the people in a society. As shown in Figure 1, a typical Alejandra appears to have a darker complexion and hair than a typical Heather, while Ethan mostly appears as a little boy since it is a recently popular male name. Taking these examples further, specific first names vary in prevalence even within a race. For example, though both of the following names are primarily Caucasian, the name "Anthony" has an Italian origin, and the name "Sean" has an Irish origin. We might ex-

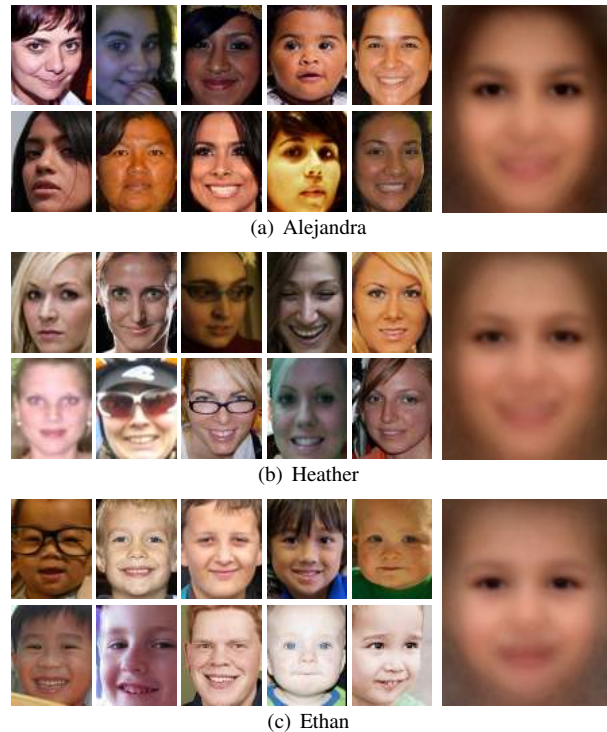

(a) Alejandra

(b) Heather

(c) Ethan

Figure 1: Face examples of 2 female and 1 male names and their average faces computed from 280 aligned faces. Comparing the average faces, Alejandra (often Hispanic) has darker skin and hair than the average face of Heather (often Caucasian). In contrast, Ethan (a popular boy's name in recent years) has a much younger looking.

pect different distributions of and correlations between facial shapes, complexions, and facial hair within even these two (primarily) Caucasian male first names. In a sense, each first name represents a joint distribution over a large set of facial attributes. In this work, we represent the appearance of many first names, and show that this name-based representation of facial appearance is a powerful face attribute.

This paper introduces, and then begins to answer, a new question in facial processing: How effectively can we infer the name of a person from only a single photograph, and with no other image examples of that face? Of course, it is unrealistic to expect highly accurate performance at

this task. After all, even identical twins each have unique names. Nevertheless, such a system, even if imperfect, could have a broad range of applications in security (e.g., finding fake person identities in a database) and biometrics (e.g., inferring the gender, age and ethnicity by guessing likely names of a face). In this paper, we show applications in constructing gender-from-names and age-from-names classifiers that achieve superior performance. One compelling advantage of our approach is that the name models can be learned using already name-tagged images from social media such as Flickr. Consequently, our gender-from-names and age-from-names classifiers do not require any additional manually labeled gender or age training data.

Our contributions are the following: First, we present the first treatment of first names as a facial attribute. Our model includes a novel matched face pyramid and Multi-Feature SVM representation, and has the advantage that all necessary images and labels are mined from the internet. Second, we show that our model is surprisingly accurate, guessing the correct first name at a rate greater than $4\times$ the expected random assignment (and greater than $2\times$ if gender is assumed to known) from a pool of 100 choices. Third, we show applications using names as attributes for state-of-the-art facial gender and age classification that require no manually labeled training images.

2. Related Work

This paper builds on recent ideas in the areas of computer vision (for numerically representing facial appearance as features) and social psychology (for investigating the social impact of first names). Here, we place more emphasis on the social psychology work because it is likely to be less familiar to the computer vision audience.

In computer vision, face detection and recognition achievements now date back around four decades [17]. Building from work showing that attributes provide good descriptions of objects [6], several papers have shown advantages in describing facial appearance in terms of a large number of attributes [18, 19] such as "male", "middle-aged", "asian". In addition to proposing a set of facial attributes, a large training set was manually labeled for each attribute at high cost ($\geq$ 10M labels total). Further, because the attributes are learned independently, the relationships and correlations between the attributes must also be modeled to improve performance. In our work, first names are treated as attributes, and the representation implicitly jointly models the age, gender, race, and other (possibly unnamed) appearance attributes associated with the people having that first name (Figure 1). Our work has a flavor similar to [3], where Berg and Belhumeur applied pairwise person classifiers to the task of face verification. Nevertheless, each of their person classifiers was trained using faces of two specific individual persons, which drastically differs from our

approach that trains models on face images sampled from first names.

In [10], names from captions are matched to the faces in the image based on attributes of age and gender (derived from facial analysis from images, and from records of name popularity over time). In this paper, we extend attributes far beyond the simple modeling of faces using gender and age attributes, to an appearance model of what distinguishes first names from one another.

Our work is the first attempt of modeling the relation between facial appearance and first names from a computer vision perspective. We build representations of faces for given names, and use these representations as an attribute description for faces. At a first glance, it might seem odd to expect that learning appearance models for different first names would be a fruitful strategy for facial appearance modeling. However, social psychology shows two important results regarding names. First, it shows that names matter and affect the lives of the people to whom they are assigned [7, 14, 16, 23, 26, 25]. Second, people themselves employ stereotypical models for names that even affect their perception of attractiveness and appearance [12, 21]. Building on the findings from social psychology studies, in this work, we also demonstrate the power of first name attributes via a series of facial analysis experiments.

Names and character traits are dependent. In [16], we see that juvenile delinquents do not have the same name distribution as the general population, even after controlling for race. Indeed, unpopular names, also correlated with a lack of education, are more common in the delinquent population. Further, [7] shows that first names associated with lower socio-economic status (e.g., names with an apostrophe, with a high "Scrabble score", or having other attributes) result in both lower standardized test scores and lower teacher expectations, even after using sibling pairs to control for race and socio-economic status.

The name a person receives at birth also affects that person's preferences and behaviors. Letters belonging to the first or last name are preferred above other letters [23]. This preference appears to transcend the laboratory and influence major life decisions. In a series of papers, Pelham, Jones, and collaborators call the effect *implicit egotism*, the gravitation towards people, places and things that resemble the self. People disproportionately choose spouses with names similar to their own [14]. For example, Eric marries Erica at a greater than the expected rate. People have careers and states of residence that are similar in sound to their names at disproportionate rates [26]. For example, Dennis is more likely to be a dentist than expected by chance, and more people with surnames beginning with Cali- live in California than expected by chance. This line of work is extended to towns of residence and street names in [25].

People have stereotypical ideas about names, and the ap-

pearance of people with those names. In one study [12], girls' photographs were rated for attractiveness. Those photos assigned desirable names (at the time, Kathy, Christine, or Jennifer) were rated as more attractive than those assigned less desirable names (Ethel, Harriet, or Gertrude) even though the photographs were ranked as equally attractive when no names were assigned. In another relevant study [21], subjects first used facial manipulation software to produce stereotypical face images for 15 common male names (e.g., Andy, Bob, Jason, Tim) by varying facial features. Additional subjects are able to identify the prototype names for each face at rates far above random guesses (10.4% vs. 6.7%) and for 4 of the 15 faces, the majority vote name was correct. This strong evidence provides motivation for us to learn, from actual images, visual models for first names.

3. Name100: A First Name Face Dataset

To model the relation between names and appearance, we assembled a large dataset by sampling images and tags from Flickr. The dataset contains 100 popular first names based on the statistics from the US Social Security Administration (SSA) [2], with 800 faces tagged for each name. The 100 names were selected as follows: First, we ranked the names from the SSA database in order of the total number of times each name was used between 1940 and the present. Then, the top names for males and females were found. In turn, first names were used as a Flickr query, and names for which enough (≥ 800) image examples were found were kept in the dataset. The completed dataset includes 48 male names, 48 female names, and 4 neutral (a name held by both males and females) names to model the real-world distribution of names. These names cover 20.35% of U.S. persons born between 1940 through 2010. We use the name as a keyword to query Flickr and enforce the following criteria when sampling images, in an effort to sample first-name appearance space as fairly as possible: First, since name ambiguities arise when multiple people are present in an image, we run a face detector [15] and eliminate those images that contain multiple faces, and check if there exists one and only one first name in the image tag. Second, we filter out images that are tagged with any of 4717 celebrity names that could bias the sampling. Without this consideration, a query of "Brad" would return many images of the movie star "Brad Pitt", and distort the facial appearance distribution of the name "Brad". Last, no more than one image is downloaded per Flickr user. This rule is meant to prevent multiple instances of a person "David", when "David" appears in many images of a particular Flickr user. While these rules may not be sufficient to prevent all instances of either incorrectly named faces, or different images of the same person appearing more than once, they are effective at preventing many problems that more naïve strategies would

encounter, and we found them to be effective.

4. Modeling the Names and Faces

Figure 2 shows an overview of the our system. First, the faces are normalized for scale with detected eye positions [5] and resampling the face to 150×120 pixels. We extract SIFT descriptors [22] by sampling on a dense grid with 2-pixel intervals. Each 128-dimensional SIFT descriptor is then encoded by the Locality-constrained Linear Coding (LLC) method [27] to a 1024-dimensional code. These encoded LLC codes are aggregated over a spatial pyramid [20] using max pooling, such that we have a 1024-dimensional vector at each of the 21 pyramid grid. This produces a feature vector of $21 \times 1024 = 21504$ dimensions for each face.

For each pair of first names, we then build a Support Vector Machine (SVM) [4] classifier to discriminate between that pair of names (more details on classifier construction are in Section 5). Therefore, classifying N names requires $\frac{N \times (N-1)}{2}$ pairwise classifiers. This 1-vs-1 classifier construction [13] is common for multi-class problems, and particularly relevant for distinguishing between first names. The visual features that distinguish any particular pair of individuals varies. For example, "David" and "Mary" differ in gender, but "David" and "Ethan" differ mainly in age ("Ethan" is a younger name). We also experimented with using a 1-vs-all approach for classifier training, and found the results to be inferior to the 1-vs-1 classifiers. Using these pairwise name classifiers, a test face can then be described by a vector of $\frac{N \times (N-1)}{2}$ dimensions, each being an SVM output score indicating whether the name of the face is more likely to be the first or the second in the name pair. We call this feature vector the *pairwise name attribute* representation of the face.

The pairwise name attributes establish the link between a face and the names that best fit its appearance, which naturally leads to many interesting applications as we describe in Section 6. We show that our system accomplishes the obvious task, guessing the first name of a person, at rates far superior to random chance, even after accounting for the effects of age and gender. We then describe an application of gender classification based on our pairwise name attributes, which achieves state-of-the art performance. Further, we demonstrate that the pairwise name attributes are very effective on the task of age classification.

It is important to point out that our entire system requires no human labeling in addition to the Flickr name tags. This gives our system several unique advantages. First, our system is inexpensive to deploy. By not requiring any additional human labels, we do not need to pay human workers and we avoid costs associated with training workers. The labels that we do use (first names tagging the images) are freely provided on the Internet because they already pro-
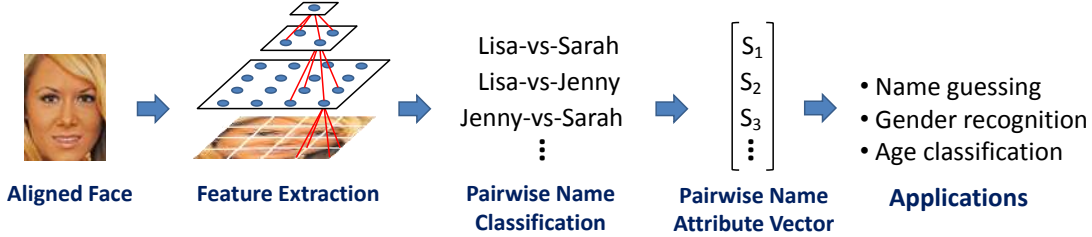
Figure 2: Overview of our system. First, a query face is represented as a 3-level pyramid of max-pooled LLC codes, with 1 pyramid grid at the top level, 4 at the next, and 16 at the bottom level. Next, the face is classified in a 1-vs-1 fashion with a set of pairwise name classifiers. The pairwise name classifiers outputs confidence scores which we call pairwise name attribute vector, which can be used for many applications as we will show Section 6.

vide value for searching and sharing the images. Second, because our system is driven by first names as attributes, we avoid semantic issues related to attribute tagging (e.g. ideas about what constitutes "attractive" vary between observers). Finally, our system is easily extensible. Although, for now, we explore the popular first names from the United States, extending the system to other cultures is as easy as performing additional image downloads with additional name queries as search terms.

5. Pairwise Name Classification using Multi-Feature SVM

As mentioned in the previous section, each face is represented as a $21 \times 1024 = 21504$ dimensional feature vector. Conventionally, as has been done in [27], this extremely high dimensional vector is directly fed to an SVM for classification. However, performing classification in such a high dimensional feature space is susceptible to overfitting, especially on our challenging classification task of assigning first names to faces. Therefore, instead of simply concatenating the 1024 dimensional LLC codes from all 21 pyramid grids, we regard a face as represented by 21 feature vectors, each vector coming from one pyramid grid. In this way, the 21 feature vectors can be viewed as coming from 21 feature channels that are complementary to each other, and we propose a method called Multi-Feature SVM (MFSVM) that effectively fuses the features together to achieve a better performance on the task of first name classification.

Our MFSVM follows the framework of AdaBoost [8], with the classifiers being SVMs working on different feature channels. To begin, we initialize equal weights on all training images and use feature channel 1 to perform a 5-fold cross validation using SVM. The misclassified training images with that SVM are given higher weights when training the SVM for feature channel 2. Intuitively, the SVM for feature channel 2 will focus on the highly weighted misclassified images from feature 1's SVM. This procedure is repeated until we have trained an SVM on each of the feature channels.

Suppose there are T feature channels and N training im-

ages, we denote the training data as $x_{t,i}$, where $t = 1, ..., T$ and $i = 1, ..., N$, meaning the t-th feature extracted from the i-th training image. Each training image is associated with a training label $y_i \in \{-1, +1\}$. For a test image, the testing data is z_t. The MFSVM can be written as follows:

Data: Training data $x_{t,i}$, training labels $y_i \in \{-1, +1\}$, testing data z_t, where $t = 1, ..., T$ and $i = 1, ..., N$
Result: SVM classifiers $f_t(z_t)$, classifier weights α_t
Initialization: weights $D_i = 1$;
for $t = 1 : T$ **do**
  (a) Using weights D, perform SVM cross validation to obtain confidence $f_t^{cv}(x_{t,i}) \in R$ and prediction $\hat{y}_{t,i}^{cv} = sign(f_t^{cv}(x_{t,i}))$, compute error $err_t = \frac{\sum_{t=1}^{N} \mathbb{I}\{\hat{y}_{t,i}^{cv} \neq y_i\}}{N}$;
  (b) Train SVM f_t using D;
  (c) Compute $\alpha_t = \frac{1}{2} \log(\frac{1-err_t}{err_t})$;
  (d) Set $D_i = D_i \exp(-\alpha_t y_i f_t^{cv}(x_{t,i}))$, and renormalize so that $\sum_{t=1}^{N} D_i = N$;
end
Output the final classifier $f_{all}(z) = \sum_{t=1}^{N} \alpha_t f_t(z_t)$
Algorithm 1: Multi-Feature SVM

In practice, we fuse the 21 features channels from coarse to fine grids on the face image pyramid. In our experiments we find that the ordering does not have much effect on the performance. On average, the pairwise name classifiers perform quite well at distinguishing between first names as shown in Table 1. As expected, it is easier to classify between names that differ in gender. We also found, within each gender, the pairs of names that are easiest and hardest to distinguish, see Table 1. Easy-to-distinguish name pairs tend to have different ages (Figure 3). Name pairs that are hard to distinguish tend to have similar popularity patterns.

6. Applications of Name Models

In this Section, we explore the performance of our pairwise name representation for a variety of tasks. We first show that the name models are surprisingly accurate on the task of first name prediction, then raise novel applications that utilize names for gender and age classification.
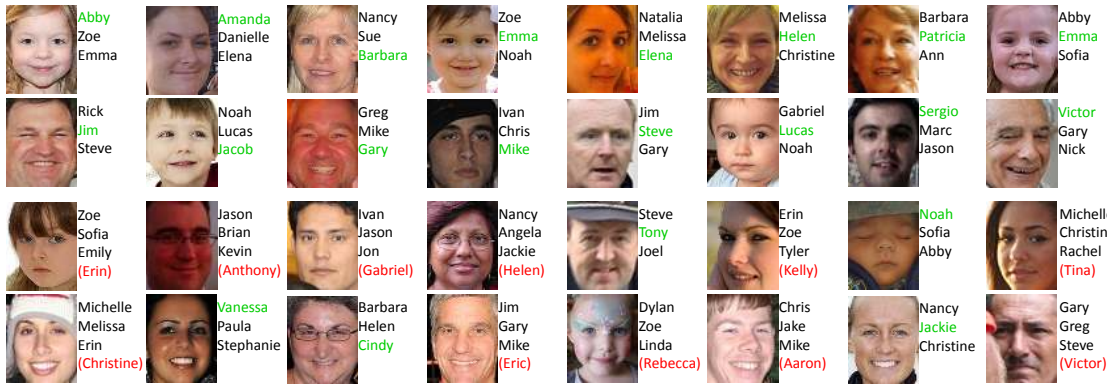
Figure 4: The top-3 predicted names for some face images. The correct prediction is highlighted in green, while the actual first name is shown in red if it is not ranked within the top-3 predictions. The first 2 rows give some good examples where our top-3 predictions include the actual name, and the bottom 2 rows are randomly selected from our test set. Even when our predictions are wrong, reasonable names are predicted (e.g., appropriate gender or age).

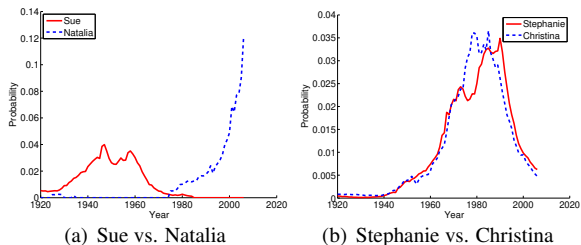(a) Sue vs. Natalia   (b) Stephanie vs. Christina

Figure 3: Probability of birth year for an easy to distinguish female pair (a) Sue vs. Natalia: accuracy of 76.3%, and a difficult to distinguish pair (b) Stephanie vs. Christina: accuracy of 46.1%.
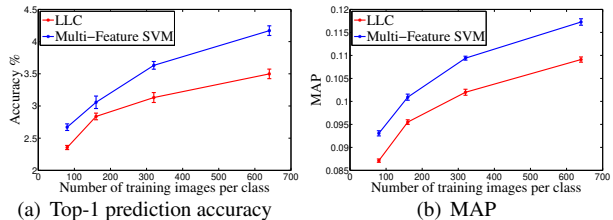

(a) Top-1 prediction accuracy   (b) MAP

Figure 5: Evaluation of first name prediction on the Name100 dataset. The task is to predict the first name of a previously unseen face from 100 choices. The results of both the MFSVM classifier and the original LLC method are far better than random guess (MFSVM accuracy 4.17% vs. random accuracy 1%, MFSVM MAP 0.117 vs. random MAP 0.052), with MFSVM showing improved performance over the LLC method.

	Accuracy	STD
Overall	69.4%	11.1%
Male-Female	79.5%	4.0%
Male-Male	59.5%	6.4%
Female-Female	59.1%	5.0%
Best Male: Noah vs. Steve	79.3%	
Best Female: Sue vs. Natalia	76.3%	
Worst Male: Mike vs. Brian	45.9%	
Worst Female: Stephanie vs. Christina	46.1%	

Table 1: A summary of the performance of the pairwise name classifiers. The top four rows summarize the overall performance at distinguishing between two names. The bottom four rows show the most and least accurate pairwise name classifiers when classifying between two mostly male or two mostly female names. Mike vs. Brian and Stephanie vs. Christina are indistinguishable to our classifier (which performs at the level of random chance) because the gender, age, and ethnic makeup of the samples with those name pairs are so similar. For all rows, random chance results in a 50% accuracy.

6.1. First Name Prediction

First name predictions are derived from the pairwise name attributes as follows: Each first name is associated with $N-1$ pairwise name classifiers. The total name margin for a particular name is produced by marginalizing over each associated pairwise name classifier. By sorting the first names according to the total name margins, a rank-ordered list of first names is produced.

We evaluate the performance of first name predictions on our Name100 dataset by 5-fold cross validation. The dataset contains 100 names $\times$ 800 faces/name $= 80,000$ faces. In each fold we test on $16,000$ faces with equal number of testing examples per name, while varying the number of training examples to study the effect of training data size on the name prediction performance. The learning curves of top-1 prediction accuracy and Mean Average Precision (MAP) are plotted in Figure 5. Our MFSVM classifiers fuse the 21 max-pooled LLC codes from the face pyramid and offer a significant performance gain over the original LLC method. With 640 training images per name, we achieve 4.17% top-1 prediction accuracy and 0.117 MAP, which is far better than the random guess performance of 1.00% accuracy and 0.052 MAP. Table 2 shows the performance of our model for guessing first names as a function of the number of names. Some examples of first name predictions are shown in Figure 4.

How is it possible that names can be guessed more than $4\times$ better than random? It is because names are not randomly distributed across people, and many correlations exist between given names and various facial features (e.g., skin color, male-ness, facial feature size, age, and possibly

Number Names N	5	10	40	70	100
Random Guess	20.0%	10.0%	2.50%	1.43%	1.00%
Our approach	**39.4%**	**23.5%**	**8.19%**	**5.41%**	**4.17%**

Table 2: Performance of our approach for guessing first names given randomly selected subsets of N names.

Method	Prediction accuracy	MAP
Our approach	**4.17%**	**0.117**
Gender → name	1.61%	0.075
Age → name	1.37%	0.063
Gender + age → name	2.33%	0.089
Random guess	1.00%	0.052

Table 3: Comparison of our approach to the methods of including gender and age effects on first name prediction. By directly modeling names and faces, we achieve much better performance even when gender and age effects are taken into account.

even nameless attributes [24]).

To more thoroughly investigate the relationship between names and faces, we examine a baseline of estimating gender and age for the task of name prediction. In other words, how accurately can we guess a first name, given only the estimated age and gender of the face? We train gender and age classifiers using the Group Image Dataset [11], a dataset which contains a total of 5,080 images with 28,231 faces manually labeled with ground truth gender and coarse age categories (age categories include 0-2, 3-7, 8-12, 13-19, 20-36, 37-65, 66+). We construct the gender and age classifiers in the exact same manner as we train the name models, by first extracting max-pooled LLC codes on the face pyramid, then passing the features to MFSVM classifiers and finally marginalizing the outputs from the classifiers. Having trained the gender and age classifiers, we use them to predict the gender and age of the faces in our Name100 dataset. The gender and age predictions associated with a testing face are not independent of first name, hence considering these features offer a better performance than random guess. First names are predicted from gender and age estimates as follows: Considering estimated gender, if a test face is classified as a male, then we make a random guess among the male names. Considering estimated age category, we compute the range of predicted birth years by subtracting the predicted age from the image taken year. Since each name has a birth year probability distribution over time (see Figure 6), the first name is predicted as the name that has the maximum birth probability within the range of predicted birth years. We can also combine gender and age, by incorporating the estimated age information to make first name guess only within the subset of names selected by the estimated gender. Table 3 compares our name models trained using 640 images/name to the baseline performances achieved by considering estimated age and gender as described above. Our name models achieve superior performance (4.17%), even versus the baseline that combines both gender and age classifiers (2.33%). This observation shows the advantage of our approach that directly constructs appearance models for first names, rather than introducing an intermediate layer of variables (e.g., gender and age) to learn the relation between names and their facial appearances. In other words, our name models capture visual cues beyond just age and gender.

We additionally evaluated the human performance on guessing first names via Amazon Mechanical Turk. The test samples include 2000 male and female face images from our Name100 dataset, and we have 3 workers work on each image. As it is unrealistic to ask human to select 1 name out of the 100 names, we show a face with 10 possible names, where the names include the correct name and 9 other random names of the same gender in random order. The human prediction accuracy is 13.7% (significantly better than the random baseline of 10%), compared to our method that achieves 18.2% accuracy within the 10 selected names, with margin of error 1.4%.

6.2. Gender Recognition From Names

Using our first name attributes, we are able to construct a state-of-the-art gender classifier by exploiting the fact that many first names have a strong association with gender. Intuitively, if a face seems more like an "Anthony" than an "Anna" then it is more likely to be the face of a male. Our gender classifier works as follows: First, we produce the pairwise name attribute vector for each test face. Next, we order the first names by their total name margins as described in Section 6.1. Finally, we classify the gender of the test face as male or female depending on the gender associated with the majority of top 5 names in the ordered list of 100 first names. A neutral name is counted as either a male or a female name based on the gender ratio of that name, which is computed with SSA database [2] statistics.

We evaluate the gender recognition performance on the Group Image Dataset [11], which contains faces with a large variation of pose, illumination and expression. As a benchmark, we used the facial attribute software web service provided by Kumar [1] for predicting gender by the method of [19]. Their facial attribute service runs its own face detector, which correctly detected 22,778 out of 28,231 faces from the Group Image Dataset, and we filtered out their falsely detected faces with the ground truth face positions. We compare the gender classification algorithms on these 22,778 test faces. As reported in Table 4, our method outperforms the result of [19], and achieves a gender classification accuracy of 90.4%, which is an impressive 29% reduction in error. It is important to again note that our gender classifier uses name models trained with names freely available on the web, and does not require *any* manually labeled gender training examples. As another comparison, we trained SVMs for gender classification on images from
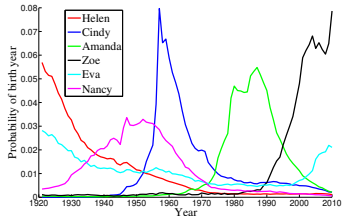
Figure 6: The birth year probabilities of a set of names, where many names show varying popularity over the years.

[11] with 2-fold cross-validation. This strongly supervised scheme achieves 89.7%, still below our 90.4%, and again, our method has the benefit of not requiring manual labels.

Algorithm	Gender recognition accuracy
Gender-from-names	**90.4%**
Kumar et. al. [19]	86.4%
Prior	52.4%

Table 4: Without any gender training labels, we perform gender recognition using our name models and achieve state-of-the-art performance.

6.3. Age Classification From Names

Due to the evolution of culture and occurrence of significant events, the popularity of a name varies over time. We use the statistics from the SSA database to plot the birth year probabilities of several names in Figure 6, where it can be seen that the birth probabilities of names have large fluctuations over the years. If a person is named Zoe, she is likely to be young because the name Zoe became popular during the 1990s. Thus, once we are able to describe a test face with our first name models, then we can utilize the birth probability of names to predict the age of the face. The advantage of such an age-from-names approach is obvious: as with our gender classifier, we again do not require any age ground truth labels to produce a reasonable age classification.

Our age-from-names classification works by first generating a ranked list of 100 names for a test face (again following Section 6.1), using the 4950 pairwise name models trained for first name prediction. We also compute the birth year probabilities from 1921 to 2010 for these 100 names, using the SSA baby name database. Certainly, the names ranked at the top of the list should be given higher weights for the task of age classification. Therefore we assign exponentially distributed weights to the ranked 100 names, such that the i-th name is associated with a weight of $\omega_i = \lambda e^{-\lambda i}$, where $\lambda = 10$. Denoting the birth probability of the i-th ranked name in year j as $p_i(j)$, then the birth probability of the ranked 100 names are combined using weighted product:

$$p_{\text{combined}}(j) = \frac{\prod_{i=1}^{100} p_i(j)^{\omega_i}}{Z} \quad (1)$$

where $Z = \sum_j p_{\text{combined}}(j)$ is a normalization term.

Each test image contains a time stamp in its JPEG metadata, so we know the year that the image was taken. Suppose that the test image was taken in the year 2008 and we believe the face falls into the age category of 20-36, then the person should be born within the year range of 1972 to 1988. We assign the confidence score for the face belonging to the age category of 20-36 as the mean of the combined birth probability over the proportion of the years 1972 to 1988. The confidence score can be written as:

$$\text{Confidence of age } t_1 \text{ to } t_2 = \frac{\sum_{j=s-t_2}^{s-t_1} p_{\text{combined}}(j)}{t_2 - t_1 + 1} \quad (2)$$

where s is year that the image was taken, t_1 and t_2 specify the lower and the upper bound of the age category respectively.

Once again, we evaluate our age classification performance on the Group Image Dataset. Equation (2) is employed to compute the confidence scores for the 7 age categories of 0-2, 3-7, 8-12, 13-19, 20-36, 37-65, 66+, as specified in the dataset. The age category with the largest confidence score will be picked as the age prediction for the test face. We work on the same test partition that was used in [11], where there are an equal number of testing instances for each age category (1050 images total). Table 5 reports the accuracy for exact category match, as well as the accuracy when an error of one age category is allowed (e.g., a 3-7 year old classified as 8-12). We benchmark our age-from-names classifier against the performance of [11], where our system shows a significant improvement. When allowing an error of one age category, our age classifier achieves 88.0% accuracy, which is surprisingly good given the fact that we are simply utilizing the age information hidden inside the names and use no other manually labeled information. While we are pleased with this performance, we do not claim to have state-of-the-art accuracy for age estimation [9], which currently relies on manifold learning and regressing using training images for which the actual age of each face is known. We do claim to be the most accurate (and only) age estimation method for which no age labels are provided for the training images.

Algorithm	Accuracy for exact match	Allow error of one age category
Age-from-names	**41.4%**	**88.0%**
Gallagher & Chen [11]	38.3%	71.3%
Random Prior	14.3%	38.8%

Table 5: We perform age classification using the birth probability of names over years 1921-2010. Without any age training labels, our age classification result shows significantly improved result compared to [11].

6.4. Name Embedding

We visually explore the visual similarity between first names to produce a first name embedding that represents the
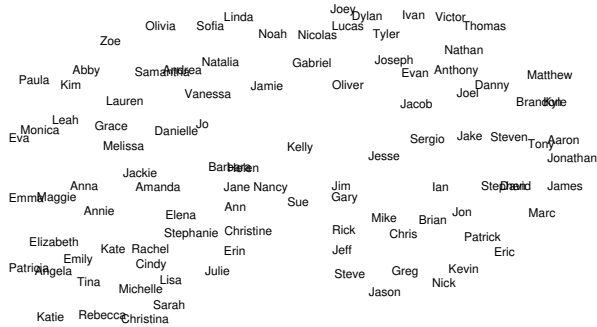
Figure 7: An embedding of first names. By analyzing the confusion between our first name classifiers and then embedding the first names into a two-dimensional space, we see that visually similar names are placed near one another.

visual similarity of the faces having the first names in our dataset. Some pairs of first names are easier to distinguish than others. In a sense, first names that are interchangeable (i.e., pairs of names that perspective parents were deciding between) should have face populations that appear to be similar, and should be close in our embedding.

To produce the first name embedding, a matrix indicating the accuracies of each of the pairwise name classifiers, after scaling and inverting so that the most confused name pair has distance 0, and the most distinct name pair has distance 1, is used as an affinity matrix for multi-dimensional scaling (MDS). Following MDS, we apply a force model to reduce the incidents of names overlapping to facilitate viewing in this paper. Figure 7 shows our embedding. Notice that the horizontal dimension relates to gender (males on the right) and age corresponds to the vertical axis (younger names are near the top). Similar names are placed nearby one another in the embedding. Again, we emphasize that this name embedding is produced solely as a by-product of our pairwise name classifiers, and is completely based on the visual similarity between faces having the given names.

7. Conclusion

In this paper, we consider a new problem of facial processing by modeling the relation between first names and faces. We build models for common names and treat first names as attributes for describing the facial appearance. A novel matched face pyramid using Multi-Feature SVM is proposed, which offers improved performance on constructing first name attributes. We show the surprising result that first names can be correctly inferred at rates far exceeding random chance. We have also described several applications of our name attributes, including first name prediction, gender recognition and age classification. Our first name attributes representation is powerful for performing various facial analysis tasks, and has the advantage of using name labels that are freely available from the internet.

References

[1] Facial attributes software. http://afs.automaticfacesystems.com/.

[2] US social security administration baby name database. http://www.ssa.gov/oact/babynames.

[3] T. Berg and P. Belhumeur. Tom-vs-Pete classifiers and identity-preserving alignment for face verification. In *Proc. BMVC*, 2012.

[4] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Trans. on Intelligent Systems and Technology*, 2011.

[5] T. Cootes, C. Taylor, D. Cooper, and J. Graham. Active shape models - their training and application. *CVIU*, 1995.

[6] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. In *Proc. CVPR*, 2009.

[7] D. Figlio. Names, expectations and the black-white test score gap. NBER working papers, National Bureau of Economic Research, 2005.

[8] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journ. of Computer and System Sciences*, 1997.

[9] Y. Fu, G. Guo, and T. Huang. Age synthesis and estimation via faces: A survey. *PAMI*, 32(11), 2010.

[10] A. Gallagher and T. Chen. Estimating age, gender, and identity using first name priors. In *Proc. CVPR*, 2008.

[11] A. Gallagher and T. Chen. Understanding images of groups of people. In *Proc. CVPR*, 2009.

[12] S. Garwood, L. Cox, V. Kaplan, N. Wasserman, and J. Sulzer. Beauty is only "name" deep: The effect of first-name on ratings of physical attraction. *Journ. of Applied Social Psychology*, 10(5), 1980.

[13] C.-W. Hsu and C.-J. Lin. A comparison of methods for multiclass support vector machines. *IEEE Trans. on Neural Networks*, 2002.

[14] J. Jones, B. Pelham, M. Carvallo, and M. Mirenberg. How do I love thee? Let me count the Js: Implicit egotism and interpersonal attraction. *Journ. of Personality and Social Psychology*, 87(5), 2004.

[15] M. Jones and P. Viola. Fast multiview face detector. In *Proc. CVPR*, 2003.

[16] D. Kalist and D. Lee. First names and crime: Does unpopularity spell trouble? In *Social Science Quarterly*, volume 90, 2009.

[17] T. Kanade. *Picture Processing by Computer Complex and Recognition of Human Faces*. PhD thesis, Kyoto University, 1973.

[18] N. Kumar, P. N. Belhumeur, and S. K. Nayar. Facetracer: A search engine for large collections of images with faces. In *Proc. ECCV*, 2008.

[19] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Describable visual attributes for face verification and image search. In *PAMI*, 2011.

[20] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proc. CVPR*, 2006.

[21] M. Lea, R. Thomas, N. Lamkin, and A. Bell. Who do you look like? Evidence of facial stereotypes for male names. *Psychonomic Bulletin and Review*, 2007.

[22] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004.

[23] J. Nuttin. Narcissism beyond gestalt and awareness - the name letter effect. *European Journ. of Social Psych.*, 15(3), 1985.

[24] D. Parikh and K. Grauman. Interactively building a discriminative vocabulary of nameable attributes. In *Proc. CVPR*, 2011.

[25] B. Pelham, M. Carvallo, T. DeHart, and J. Jones. Assessing the validity of implicit egotism: A reply to Gallucci. *Journ. of Personality and Social Psychology*, 85(5), 2003.

[26] B. Pelham, M. Mirenberg, and J. Jones. Why Susie sells seashells by the seashore: Implicit egotism and major life decisions. *Journ. of Personality and Social Psychology*, 82(4), 2002.

[27] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *Proc. CVPR*, 2010.