Compare and Contrast: Learning Prominent Visual Differences

Steven Chen   Kristen Grauman
The University of Texas at Austin

Abstract

Relative attribute models can compare images in terms of all detected properties or attributes, exhaustively predicting which image is fancier, more natural, and so on without any regard to ordering. However, when humans compare images, certain differences will naturally stick out and come to mind first. These most noticeable differences, or prominent differences*, are likely to be described first. In addition, many differences, although present, may not be mentioned at all. In this work, we introduce and model prominent differences, a rich new functionality for comparing images. We collect instance-level annotations of most noticeable differences, and build a model trained on relative attribute features that predicts prominent differences for unseen pairs. We test our model on the challenging UT-Zap50K shoes and LFW10 faces datasets, and outperform an array of baseline methods. We then demonstrate how our prominence model improves two vision tasks, image search and description generation, enabling more natural communication between people and vision systems.*

1. Introduction

Suppose you are asked to compare and contrast two different shoes, shown in Figure 1. You might say that the left shoe is *more formal* than the right shoe, then perhaps state that the left shoe is *more shiny* and *less comfortable* than the right shoe. As soon as you are given the images, these differences stick out and are most noticeable. However, consider that the two shoes have a huge number of differences. For instance, the left shoe is *more rugged* than the right shoe, and also *darker*. Although these other differences are certainly present and true, they are much less noticeable to us, and we would likely mention them later, or not at all.

In general, when we perform any comparison task on a pair of images, certain differences stick out as being most noticeable out of the space of all discernible differences. These most noticeable differences, or *prominent differences*, stand out and would be described first, while most other differences are not as noticeable and would typically not be mentioned in a description.

Figure 1: When we compare images, certain differences stick out over others. Although all the attribute statements on the left are true, the prominent differences in bold stand out and would often be described first, whereas the others may not even be mentioned.

In this work, we introduce and learn prominent differences in images, expressing them through relative attributes. When people compare images, they can describe differences in their *attributes*, human-nameable visual properties of images [9, 14, 20, 24, 26, 34, 43] used to describe anything from materials (*smooth*, *furry*) and parts (*has leg*, *has glasses*) to styles (*sporty*, *formal*) and expressions (*smiling*, *sad*). *Relative attributes*, or attributes that indicate an image's attribute strength relative to other images, provide an intuitive and meaningful representation for visual comparison, and have been widely used for vision tasks [7,20,29,37,40,41,45–47]. Relative attributes express comparisons of attribute strength (*e.g.*, image X is *smiling more* than Y, but *smiling less* than Z), and are the natural vocabulary of our proposed prominent differences.

Prominent differences have many practical applications in vision. Humans interact with vision systems as both users and supervisors, and naturally communicate prominence. For instance, in an interactive search task, where humans provide comparative feedback (*e.g.*, I would like to see images like this shoe, but *more formal* [20]), the attributes that people elect to comment on are prominent differences. In zero-shot learning with relative attributes [14, 34], where humans describe unseen visual categories to a machine by comparing with seen categories, prominence could enhance learning by better understanding these comparisons. Prominent differences are the properties humans provide first when making comparisons, and thus directly influence how humans interpret comparison descriptions. Prominence could also be used to highlight the differences that

(a) dark hair (>)   (b) smiling (<)   (c) forehead (>)

(d) colorful (<)   (e) fancy (>)   (f) formal (<)

Figure 2: Different attributes stand out as prominent for different image pairs. According to the consensus of seven MTurk judges, even though 2a and 2b both differ in darkness of hair, *dark hair* sticks out as most prominent in 2a but not in 2b. The wide color difference makes *colorful* prominent in 2d, while in 2f, a combination of properties result in *formal* as the prominent difference.

stick out to people between fashion styles, extending recent work that uses attributes for fashion description [1, 11, 26].

Modeling prominent differences is challenging due to several key reasons. First, there is a large variety of reasons why an attribute stands out as prominent for any image pair. For instance, large differences in attribute strength can play a role (Figure 2d), as well as absence of other significant differences (Figure 2a) and unusual occurrences (Figure 2c). In general, complex interactions between the attributes of two images cause certain differences to stand out. Second, humans use a large and diverse vocabulary when expressing prominence, which a model should support. Finally, prominent differences are observed between individual images. As we will show, simply predicting prominence based on the prior frequency of usage for the attribute words is not sufficient. Thus, prominent differences must be modeled at the image instance level.

In this work, we propose to model prominent differences. We collect a novel dataset of prominent difference annotations, propose a model based on relative attribute features learned with deep spatial transformer networks or large margin rankers, and evaluate on two unique and challenging domains: the UT-Zap50K shoes dataset [46] and the LFW10 faces dataset [38]. We show that our model significantly outperforms an array of baselines for predicting prominent differences, including an adaptation of the state-of-the-art binary attribute dominance approach [43]. Finally, we demonstrate how our prominence model can be used to enhance two vision applications: interactive image search and description generation.

2. Related Work

Attributes   Attributes are semantic and machine-understandable properties (*e.g.*, *smiling*, *shiny*) that are used by people to describe images [1–4, 6–9, 11, 14, 17, 19, 20, 23, 24, 26, 27, 29, 34, 35, 37–39, 43, 46, 47]. Attributes serve as expressive mid-level features for recognition [24, 27, 35]. Attributes have also been used as a vocabulary for learn-

ing unseen visual categories, known as zero-shot learning [14, 25, 34, 43]. Attributes are well suited for applications in fashion, where they have been used to forecast style popularity [1], organize outfits by style [11], and drive interactive search [49]. Recently, deep convolutional neural networks (CNNs) have shown improved attribute prediction accuracy over previous approaches in various fields such as faces [27] and fashion [26]. In contrast to previous work, which focuses on detecting the presence of attributes in images, we learn which particular attribute differences stick out when comparing images.

Relative Attributes   Relative attributes, first introduced in [34], represent an image's attribute strength with respect to other images [7, 10, 20, 29, 37, 40, 41, 45–48], and are a richer representation than binary presence/absence. Relative attributes enable *visual comparisons* between images (*e.g.*, the left shoe is *more sporty* than the right), and have been used to discern fine-grained differences [46, 47] and predict image virality [7]. Recently, deep CNNs have been used to both predict relative attributes [40, 41, 45] as well as generate synthetic images of varying attribute strengths [44, 48]. However, no prior work considers which relative attributes stand out, or what relative attributes humans tend to use in speech. Our work introduces prominent differences, a novel functionality representing most noticeable differences in the vocabulary of relative attributes.

Importance of Objects and Attributes   Different concepts of visual importance have used attributes [2, 8, 42, 43]. Attributes have been used to learn object importance (defined in [42] as the likelihood an object is named first), which can be used to help rank image aesthetics [8, 18]. As opposed to objects, we consider the separate concept of prominent differences, which are selected from a vocabulary of linguistic properties.

Turakhia and Parikh [43] introduce the related concept of binary attribute dominance, measuring which binary attributes (*e.g.*, *is 4 legged*, *not smiling*) stand out more for different object categories. Our work is distinct: we learn which *relative differences* stick out *between two images*, while [43] learns which *binary attributes* are more apparent *by category*. For instance, given the category of sneakers, [43] may detect general trends of *sporty* and *comfortable*; however, our prominence approach captures important differences between image instances, such as one specific shoe being prominently *more rugged* than another. We later extend binary attribute dominance and compare with our approach (*cf*. Section 5.2).

Image Saliency   Works modeling saliency (*e.g.*, [13, 16, 33]) have used attributes to predict what regions people tend to look at in images. Although saliency may have an influence on prominence, it refers to low-level regions in single

images, whereas prominence is a linguistic, pairwise concept, and the result of a combination of mid-level cues.

Image Search   Image search has benefited from attribute-based approaches [20,39,43,47]. Attribute queries have been used in search [39], and improvements have been made using binary attribute ordering [43]. We use relative attribute ordering, and apply to the interactive Whittle-Search [20,21]. Whereas previous work solicits richer user feedback for WhittleSearch [47], we leverage the implicit prominence choices users make to obtain better results with no additional human input or interface changes.

Describing Images   As semantic properties, attributes are well-suited for visual description [9,17,23,31,34,37,43]. Recent work uses attributes to generate binary attribute descriptions [9,23,43]. Attributes have been used to generate referring expressions [17,31], phrases identifying specific objects in an image. In comparison, our work focuses on attribute differences, which could be used to improve referring expressions. Works have generated image descriptions using relative attribute comparisons [34,37]. However, these methods list exhaustive statements in arbitrary order; we improve these by focusing on differences that are prominent and natural to state.

3. Approach

First, we present an overview of relative attribute models (Section 3.1). Next, we introduce our approach for modeling prominent differences (Section 3.2). Finally, we detail how we gather our prominence datasets (Section 3.3).

3.1. Background: Relative Attribute Models

Relative attributes are semantic visual properties that represent the strength of an attribute in an image relative to other images [34]. While binary attributes represent just the presence or absence of a property (*e.g.*, *is smiling*, *is not smiling*), relative attributes rank images based on their attribute strength scores (see Figure 3), and are thus well suited for visual comparison.

We now describe a general framework for relative attribute rankers. Suppose we have a set of images $I = \{i\}$, along with a vocabulary of M relative attributes $A = \{a_m\}_{m=1}^{M}$. Let $x_i \in \mathbb{R}^D$ represent the image's D-dimensional descriptor, which could be comprised of GIST [32], color, part descriptors, CNN features, or just raw pixels. Given an image pair $y_{ij} = (x_i, x_j)$, the goal of the ranker is to determine if one image contains more of a_m than the other, or if both images have similar strengths of a_m.

Relative attribute models use sets of labeled image pairs for supervised learning [10,29,34,40,41,45]. The model is given a set of ordered pairs $O_m = \{(i,j)\}$ and a set of unordered pairs $S_m = \{(i,j)\}$ such that $(i,j) \in O_m \implies i >$

less sporty   more sporty

less smiling   more smiling

Figure 3: Relative attributes can rank images across a range of strengths.

j, *i.e.*, i contains more of a_m that j, and $(i,j) \in S_m \implies i \sim j$, *i.e.*, i and j have similar strengths of a_m.

Relative attribute models learn a ranking function $\mathcal{R}_m(x_i)$ for each attribute a_m to best satisfy the constraints:

$$\forall (i,j) \in O_m : \mathcal{R}_m(x_i) > \mathcal{R}_m(x_j) \qquad (1)$$
$$\forall (i,j) \in S_m : \mathcal{R}_m(x_i) = \mathcal{R}_m(x_j). \qquad (2)$$

Different learning objectives are used to quantify constraint satisfaction, such as a wide margin classification objective [34] for an SVM [15] ranker, or a RankNet objective [5,40,41,45] for a deep CNN ranker. We experiment with both such models in our implementation.

Ranking SVM   Ranking SVM rankers optimize $\mathcal{R}_m^{(svm)}(x_i) = w_m^T x_i$ to preserve ordering while maximizing distance between closest points when projected onto w_m. $w_m \in \mathbb{R}^D$ is the weight vector to be learned, and is linear here; nonlinear models are also possible using kernels. Ranking SVMs have seen wide use for relative attributes [7,10,20,29,34,46].

CNN Ranker   Deep CNN based rankers have emerged as a strong alternative for predicting relative attributes [40, 41,45]. These models generally use a CNN optimized for paired ranking loss [5]. Compared to the Ranking SVM models, they typically achieve better accuracy but require more time for training and larger quantities of training data. We use Singh and Lee's Siamese network [40] in our work, where each branch consists of a spatial transformer network and ranker network.

3.2. Modeling Prominent Differences

We now introduce our approach for modeling prominent differences, defined as the attribute a person would mention first when comparing two given images. We first introduce a naive method, then present our approach.

Naive Widest Relative Difference   Given an image pair y_{uv}, a simple approach for predicting prominence would be to directly compare their relative attribute scores $r_m^u = \mathcal{R}_m(x_u)$ and $r_m^v = \mathcal{R}_m(x_v)$. After normalizing these scores, one can compute the relative difference $|r_m^u - r_m^v|$,

for how different the images are in terms of a_m. By taking the maximum

$$\mathcal{W}^{uv} = \arg\max_m |r_m^u - r_m^v| \qquad (3)$$

over all attributes, we obtain the naive *widest relative difference* $\mathcal{W}^{uv}$.

We hypothesize $\mathcal{W}^{uv}$ alone is inadequate for predicting prominence. As illustrated in Section 1, there are several contributing factors to prominence, such as unusual occurrence of attributes, and many other complex interactions between the properties of the image pair. In the next section, we describe our approach for modeling prominence, which uses a novel representation to learn the interactions that cause prominence. Later, we demonstrate the efficacy of our approach through comparison with widest differences and other methods (*cf.* Section 5.2).

Our Approach Suppose we have a set of images $I = \{x_i\}$, along with M relative attributes $A = \{a_m\}_{m=1}^M$ as defined before. In addition, for each a_m, we are given a set of unordered prominence image pairs $U_m = \{(x_i, x_j)\}$ such that the most prominent difference between x_i and x_j is a_m. Note that U_m is distinct from O_m and S_m, the relative attribute pairs used to train the relative attribute ranker.

Our goal is to construct a model that, given a novel pair of images $y_{uv} = (x_u, x_v)$, predicts which attribute $\mathcal{A}^{uv}$ is the most prominent difference for that image pair.

First, in order to represent y_{ij} as an unordered pair, we need a symmetric transformation $\phi(y_{ij}) = \phi(y_{ji})$ that combines the attributes of the images into a joint representation, such that the model always predicts the same prominent difference for each specific pair. The representation should also capture the wide variety of factors for why certain properties stand out as prominent, so that the model may effectively learn.

To create our feature representation ϕ, we first compute the relative attribute scores $r_m^i = \mathcal{R}_m(x_i)$ for both images x_i in the pair and all attributes in the vocabulary, using the models described in Section 3.1, resulting in scores $r_1^i, \ldots, r_M^i$ for each image. We then compute ϕ as the average of the pair's scores for each attribute, and concatenate the absolute difference between the pair's attribute scores, creating a feature vector of length $2M$:

$$\phi(y_{ij}) = \left(\frac{r_1^i + r_1^j}{2}, \ldots, \frac{r_M^i + r_M^j}{2}, \left|r_1^i - r_1^j\right|, \ldots, \left|r_M^i - r_M^j\right|\right). \qquad (4)$$

This feature representation captures the individual relative attribute properties while maintaining symmetry: for instance, unordered pair scores for each attribute a_m can be reconstructed from two feature vector components $\phi(y_{ij})_m \pm \frac{1}{2}\phi(y_{ij})_{M+m}$. We standardize attribute scores to zero mean and unit variance before they are input into $\phi(y_{ij})$.
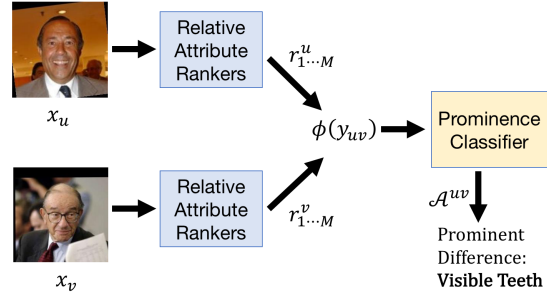

input: $y_{uv} = (x_u, x_v)$

Figure 4: Our approach pipeline for prominent difference prediction.

We experimented with other ϕ transformations on the attribute scores of an image pair, including element-wise products, absolute difference, and weighted averages. We select the given formulation of ϕ due to its strong performance in practice.

Given this representation, we now build M predictors $\mathcal{P}_m(y_{uv})$ for $m = 1, \ldots, M$ such that $\mathcal{P}_m(y_{uv})$ is the predicted confidence score that the prominent difference for y_{uv} is a_m. We predict the prominence confidence for each attribute a_m using

$$\mathcal{P}_m(y_{uv}) = \mathcal{S}_m(w_m^T \phi(y_{uv})) \qquad (5)$$

where w_m^T are the weights learned of a binary linear classifier, and $\mathcal{S}_m$ is a function mapping classifier outputs to confidence scores.

To learn the classifier weights w_m^T for each a_m, we mark all training pairs from U_m as positive examples, and all other training pairs as negatives. We use a linear SVM classifier for each attribute, due to its strong performance in practice. We use Platt's method [36] ($\mathcal{S}_m$) to transform each classifier output into posterior probabilities.

Using our full predictors $\mathcal{P}_{1 \ldots M}$, we predict the most prominent difference $\mathcal{A}^{uv}$ for y_{uv} by choosing the attribute with the highest confidence:

$$\mathcal{A}^{uv} = \arg\max_m(\mathcal{P}_m(y_{uv})). \qquad (6)$$

In addition, we can also return the top k prominent differences by selecting the k attributes with the highest confidence scores. This can be used to generate a description for a pair of images describing their k most prominent differences (*cf.* Section 4.2).

Our model follows the structure of a one vs. all multiclass classifier, with our relative attribute features $\phi(y_{uv})$ as input features and the prominent difference as class labels. Other models could certainly be considered, such as a one vs. one classifier, or a ranker. We choose the one vs. all classifier over a one vs. one for several reasons: its strong prediction performance, its easy interpretability for individual attributes, and its efficiency (only requiring one classifier per attribute). We choose a classifier approach vs. a

ranker for its ease of collecting natural human perception of prominence (see the next section), as opposed to exhaustive comparisons between all combinations of attributes, which is less intuitive and can lead to noisier results. Deep models could certainly also fit into our framework, although they require substantially more manual annotation.

3.3. Annotating Prominent Differences

No dataset exists for training or evaluating prominent differences, so we collect human annotations of prominence for image pairs at the instance level using Mechanical Turk.

To collect human perception of prominent differences, we first create a large and diverse vocabulary of M attributes that generally stuck out to annotators when viewing the dataset (see Section 5.1 for vocabulary details). We then show an annotator a pair of randomly selected images, along with a list of all M attributes $\{a_m\}, m \in \{1, \ldots, M\}$, and ask which attribute out of the list sticks out as the most noticeable difference for that image pair.

It is important to highlight that we ask each annotator to select *just one* prominent difference. This allows annotators to provide their natural first impression. Additionally, we provide the entire vocabulary of M attributes to choose from, which aids in ensuring that at least a subset of choices are noticeably different for almost all image pairs.

In addition, our approach is scalable: it requires only one annotation question per image pair, regardless of the number of attributes in the vocabulary, vs. $\binom{M}{2}$ combinations of attribute pair questions required to annotate one instance of binary dominance in [43]. This helps us scale to the instance level and capture fine-grained information on which specific images and features lead to prominence, whereas [43] collects data at the category-level, projecting the same values to all instance images in a category and losing valuable instance-level characteristics in the process.

To obtain our prominent difference ground truth labels, we collect annotations from seven annotators for each image pair. Then, for each pair, we rank the attributes by frequency chosen, and label the highest ranked attribute as the ground truth most prominent difference.

4. Applications of Prominent Differences

We now present our approaches for applying prominent differences to two human-centric applications, image search (Section 4.1) and description generation (Section 4.2).

4.1. Image Search

First, we consider applying prominent differences to WhittleSearch [20, 21], an interactive image search framework where users provide relative attribute feedback through comparisons (*e.g.*, I would like images that are *more formal* than reference image X).
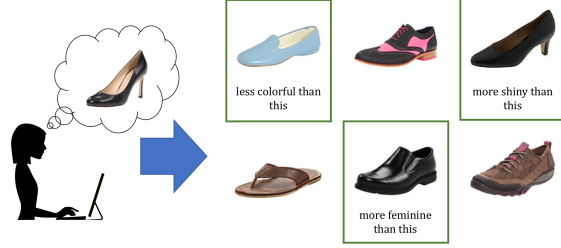

Figure 5: In WhittleSearch [20, 21], a user chooses reference images and constraints from the search page. We hypothesize that users will provide *prominent differences* between the reference and their target as feedback, and show how the proposed method can improve search.

WhittleSearch intersects the relative attribute constraints $c_1, \ldots, c_n$ provided by the user, ranking database images by how many constraints they satisfy. In each search iteration, the user is shown a page of top ranked images and selects reference images and relative attribute constraints on those images. WhittleSearch then adds the new feedback to the set of all constraints and ranks images accordingly.

When users provide feedback in the form of "What I am looking for is *more/less* a_m than image x_{ref}", by definition, they will provide prominent differences between the reference image x_{ref} and their mental target (see Figure 5). Thus, images are more likely to be relevant if they are *prominently* different in a_m with x_{ref}. We model this by introducing a relevance term p for each database image x_i using prominence:

$$p(x_i \mid c_1, \ldots, c_n) \propto \prod_c \mathcal{P}_{m_c}(x_i, x_{ref_c}), \qquad (7)$$

where $\mathcal{P}_{m_c}$ is our prominence predictor for m_c, and attribute m_c and reference ref_c are the constraint parameters from constraint c, for all $c_1, \ldots, c_n$.

Then, we rank images *within each group satisfying the same number of constraints* by listing them in descending order of $p(x_i)$. We use this approach to maintain the overall constraint ordering of WhittleSearch, while using prominence to significantly improve the ordering of images that share the same number of satisfied constraints.

A strength of our approach is that it does not require any additional user input: we simply use the user's existing feedback for prominence data. As we will show, our approach is especially impactful in the first several iterations of search, when many images satisfy all or most of the feedback constraints, and would be otherwise randomly ordered by the existing WhittleSearch algorithm [20].

4.2. Description Generation

For our second application, we consider applying prominent differences to generate textual image descriptions. Given a novel image pair, we want to generate a description comparing the images in terms of their attributes (*e.g.*, Image X is *more sporty* and *less formal* than Image Y).

When humans compare images, by definition, they state prominent differences first. In addition, humans will not name all differences for a pair; instead, they will usually describe a subset of the prominent differences. Previous work [34, 37] generates descriptions comparing images in terms of all attributes, in an arbitrary order. We argue that this is not sufficient; listing out all differences is too lengthy, while listing a random subset can miss key differences.

We propose generating descriptions containing prominent differences. Namely, given a novel image pair y_{uv}, we sort all attributes in descending order of their prominence scores, and generate a description with the top k prominent differences. For example, given two shoe images, our model can generate the description "The left shoe is *more sporty*, *less stylish*, and *less shiny* than the right shoe," stating the three most prominent differences between the instance images.

5. Results

We first introduce the two datasets we use (Section 5.1), followed by the baselines that we compare our approach to (Section 5.2). Finally, we evaluate our approach on prominence prediction (Section 5.3), as well as on image search (Section 5.4) and description generation (Section 5.5).

5.1. Datasets

We now introduce the two datasets used in our experiments, then highlight annotator agreement statistics.

UT-Zap50K Shoes Dataset   The UT-Zap50K Dataset [46] is a dataset of 50,025 shoe images from Zappos. We use a vocabulary of ten relative attributes for our experiments: (1) *sporty*, (2) *comfortable*, (3) *shiny*, (4) *rugged*, (5) *fancy*, (6) *colorful*, (7) *feminine*, (8) *tall*, (9) *formal*, (10) *stylish*. These were selected from data collected in [48], in which annotators were asked to provide the first difference that comes to mind. We use the 19 provided categories (*e.g.*, AnkleBoots, OxfordsShoes, *etc.*) as categories for the binary dominance baseline [43].

We randomly sample 2,000 images, and collect prominence for 4,990 sample pairs. For the SVM ranker and binary dominance, we generate CNN features from the fc7 layer of AlexNet [22]. For the deep CNN ranker [40], the inputs are raw pixels. For prominence prediction, we report the average of 10-fold cross validation. Images used in training pairs are disjoint from images used in testing pairs.

LFW10 Faces Dataset   The LFW10 Dataset [38] is a collection of 2,000 face images from LFW [12], along with 10,000 relative attribute annotations over ten different attributes, (1) *bald head*, (2) *dark hair*, (3) *eyes open*, (4) *good looking*, (5) *masculine*, (6) *mouth open*, (7) *smiling*, (8) *visible teeth*, (9) *visible forehead*, (10) *young*. We use these attributes as our vocabulary, and create categories for binary

dominance [43] by matching images to their subjects. We keep all individuals with three or more instances, resulting in 1,064 images from 150 categories.

We collect prominence for 1,463 sample pairs. For image descriptors for the SVM rankers and binary dominance, we use the 8,300 dimension part features learned on dense SIFT [28] provided by [38]. We reduce the dimensionality to 200 using PCA to avoid overfitting. We report the average of 5-fold cross validation.

Annotator Agreement   77% of Zap50K image pairs had three or more annotators out of seven agree on the most prominent difference, with 87% for LFW10. On average, 3.8 unique attributes were chosen as most noticeable for each image pair for Zap50K, with 3.3 for LFW10. (See Supp for additional attribute frequency statistics.) This high level of agreement shows that prominent differences are in fact consistent for most comparisons.

5.2. Baselines

Binary Attribute Dominance [43]   Our first baseline is Turakhia and Parikh's binary attribute dominance [43]. We follow the authors' approach as closely as possible, collecting separate dominance and binary attribute annotations for training the model (see Supp for details). We convert relative attributes into binary equivalents (*e.g.*, *sportiness* becomes *is sporty* and *is not sporty*), and extend their model to handle prominence by first computing binary dominance for each attribute and image in a pair, then selecting the attribute with the highest score for an individual image as the predicted prominent difference.

Widest Relative Difference + tf-idf   This baseline uses the (standardized) widest relative difference $\mathcal{W}^{uv}$ described in Section 3.2 to estimate prominence. We add tf-idf weighting to this baseline, which we find is a strict improvement over no weighting scheme.

Single Image Prominence   This baseline is trained on the same prominence labels as our approach, but projects the labels onto the two images of each pair. It then trains a multiclass SVM with individual images as inputs, as opposed to our approach, which trains on pairs and a pairwise feature representation.

Prior Frequency   The final baseline we use is a simple "prior frequency" model, which predicts prominent differences proportionally according to their frequency of occurrence in the ground truth.

5.3. Prominent Difference Evaluation

We evaluate prominent difference prediction accuracy, with each model predicting a single most prominent difference for each image pair. Recall that seven annotators supply ground truth prominence on each image pair. Be-
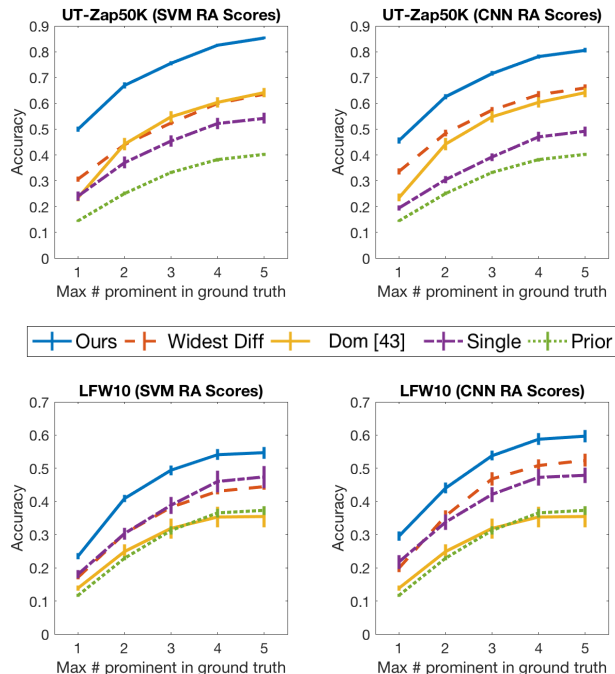
Figure 6: Prominence prediction accuracy results for two datasets (top, bottom) and either SVM or CNN-based attribute rankers (left, right).

(a) **color** ($>$), sporty, comfort

(b) **sporty** ($>$), color, comfort

(c) **feminine** ($>$), comfort, shiny

(d) **teeth** ($>$), mouth open, smiling

(e) **bald head** ($<$), dark hair, teeth

(f) **dark hair** ($<$), mouth open, smiling

Figure 7: Prominence predictions made by our approach. Predicted most prominent attribute in bold, followed by next two most confident attributes.
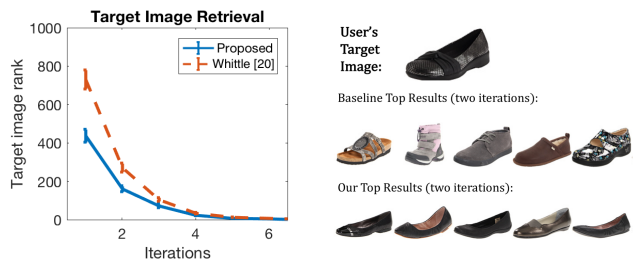

Figure 8: Image search results. Quantitative results on left, lower rank is better. On right, qualitative search results with a user's target image, followed by baseline results and our results. Our approach returns significantly more similar images to the user's mental target.

cause there is not always a unanimous prominent difference, we evaluate accuracy over a range of k maximum attributes marked as ground truth correct, to account for variance in perception. Specifically, we sort attributes by their frequency chosen, creating a partial ranking of c attributes, and take the $\min(k, c)$ most chosen as the prominence ground truth. We mark a pair as correct if the prediction $\mathcal{A}^{uv}$ is present in the ground truth. At $k = 1$, only the most prominent attribute is considered correct.

Figure 6 shows accuracy results. We divide results for both Zap50K and LFW10 into two plots, one for each attribute ranker used (ranking SVM and deep CNN). Our approach significantly outperforms all baselines for prediction. We observe sizable gains of roughly 20-22% on Zap50K, and 6-15% on LFW10 over the strongest baselines. This clearly demonstrates the advantage of our approach, which uses pairwise relative attribute features to learn the interactions between visual properties that result in prominent differences.

Our results show that the baselines are not adequate for predicting prominent differences. For widest relative difference, its lower accuracy demonstrates that strength difference is only one contributing factor to prominence: our approach is able to effectively capture other important causes. We also outperform binary dominance [43] significantly. The weak performance of the single image prominence baseline demonstrates that prominence is a pairwise phenomenon, requiring both images for context.

Comparing the use of the two attribute rankers, both yield similar performance on Zap50K but we benefit from the CNN ranker scores on LFW10. The advantages of our approach hold whether using SVM rankers or deep rankers. It is important to highlight that our contributions are orthogonal to the choice of attribute ranker: our approach can learn from relative attribute scores generated from any model.

Figure 7 shows qualitative examples. For example, the shoes in 7a are very different in many properties; despite this, our model accurately predicts *colorful* as the most prominent difference. Although the images in 7d are of the same person, our model is able to accurately predict *visible teeth* as the most prominent difference. (See Supp for more examples.)

5.4. Image Search Results

For our image search application on WhittleSearch [20, 21], we evaluate a proof-of-concept experiment using the Zap50K dataset. We use Zap50K for its large size, and sample 5,000 unseen images as our experimental database.

Due to the cost of obtaining human feedback for each combination of image and results list, we generate feedback automatically following [20]. A random subset of images from the top results page is chosen as reference images. For the user's feedback between the target x_t and each reference image x_{ref}, the user selects the most prominent difference $\mathcal{A}^{t,ref}$ to provide feedback upon. To simulate variance in
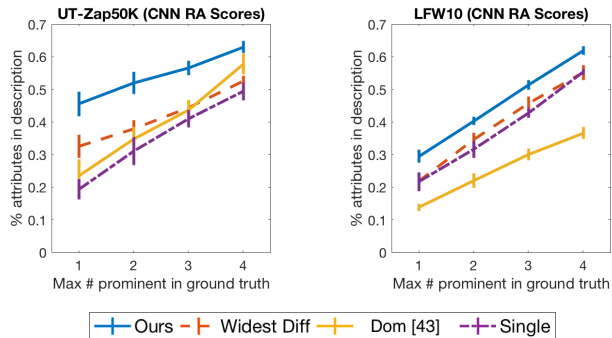
Figure 9: Description generation offline results.

(a) Left is less shiny, less formal, and more colorful than the right.

(b) Left is less feminine, more rugged, and less shiny than the right.

(c) Left has less dark hair, more bald head, and more mouth open than the right.

(d) Left is more masculine, less smiling, and less visible teeth than the right.

Figure 10: Sample generated descriptions by proposed approach.

Zap50K	Ours:	69%	Baseline:	31%
	Ground Truth:	69%	Baseline:	31%
LFW10	Ours:	61%	Baseline:	39%
	Ground Truth:	70%	Baseline:	30%

Table 1: Description generation human study results.

human perception, we add noise by randomly selecting 75% of feedback as prominent differences, and 25% as random true differences. We select 200 images as the user's mental targets. At each iteration, the user is shown the top 16 results, selects 8 images as references, and provides 8 feedback constraints using the references.

Figure 8 shows the results (more in Supp). Our approach substantially improves the target image rank over the first several iterations of search, and returns significantly more relevant images, without requiring any additional feedback.

5.5. Description Generation Results

We evaluate generated descriptions in one offline and one online experiment. For the offline experiment, we output the top k most prominent attributes that would be in a generated description, and check what percentage of the k ground truth prominent attributes are present, a metric that is critical to description quality. We compare our approach to the three strongest baselines and report results with the CNN attribute ranker in Figure 9 (see Supp for similar results with other ranker). Our approach outperforms all baselines, generating descriptions capturing more human-perceived prominent differences.

For the online experiment, we ask annotators to judge generated descriptions. Specifically, we present an image pair and two descriptions—our description of predicted prominent differences, and a baseline description with randomly chosen *true* differences—and ask which description is more natural and appropriate. We sample 200 pairs from Zap50K and 100 pairs from LFW10, generate descriptions with three statements each, and have seven judges provide feedback per pair, taking the majority vote.

For Zap50K, 69% of people preferred our description, compared to 31% for the baseline, with a p-value < 0.0001, with 61% and 39% for LFW10, with a p-value of 0.01, respectively. We also ran the same experiment using annotator ground truth prominence vs. the same baseline: people preferred the ground truth description 69% of the time for Zap50K and 70% for LFW10 (see Table 1). Our generated Zap50K descriptions are closer to the ground truth
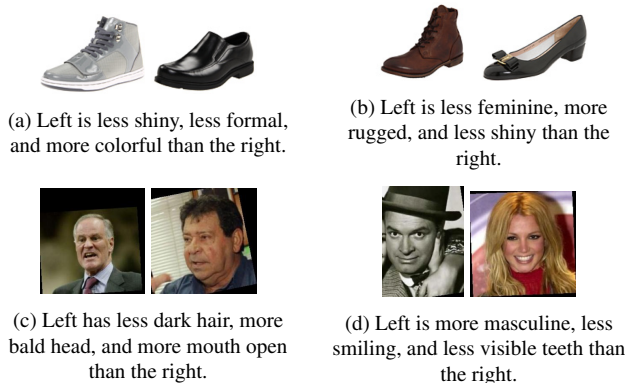
performance than for LFW10 due to our method's higher prominence prediction accuracy on Zap50K. These results demonstrate that describing images using prominent differences results in significantly more natural descriptions.

6. Conclusion

We introduced prominent differences, a novel functionality for comparing images. When humans compare images, certain prominent differences naturally stick out, while others, although present, may not be mentioned. We present a novel approach for modeling prominence at the image pair level. Experimental results on the UT-Zap50K and LFW10 datasets show that our proposed approach significantly outperforms an array of baseline methods for predicting prominent differences. In addition, we demonstrate how prominence can be used to improve two applications: interactive image search and description generation.

There is strong potential for future work. In zero-shot learning using relative attributes [34], where a human supervisor teaches a machine about an unseen visual category using comparisons, humans will provide prominent differences as supervision. This knowledge, if integrated, could result in improved classification without requiring any additional human effort. In addition, prominent differences could be used to improve referring expressions [30, 31], phrases identifying specific objects in an image. Prominent differences could be used to identify best differences to help distinguish one object over others.

References

[1] Z. Al-Halah, R. Stiefelhagen, and K. Grauman. Fashion Forward: Forecasting Visual Style in Fashion. In *ICCV*, 2017.

[2] A. Berg, T. Berg, H. Daume, J. Dodge, A. Goyal, X. Han, A. Mensch, M. Mitchell, A. Sood, K. Stratos, and K. Yamaguchi. Understanding and Predicting Importance in Images. In *CVPR*, 2012.

[3] T. Berg, A. Berg, and J. Shih. Automatic Attribute Discovery and Characterization from Noisy Web Data. In *ECCV*, 2010.

[4] S. Branson, C. Wah, F. Schroff, B. Babenko, P. Welinder, P. Perona, and S. Belongie. Visual Recognition with Humans in the Loop. In *ECCV*, 2010.

[5] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to Rank using Gradient Descent. In *ICML*, 2005.

[6] H. Chen, A. Gallagher, and B. Girod. Describing Clothing by Semantic Attributes. In *ECCV*, 2012.

[7] A. Deza and D. Parikh. Understanding image virality. In *CVPR*, 2015.

[8] S. Dhar, V. Ordonez, and T. L. Berg. High Level Describable Attributes for Predicting Aesthetics and Interestingness. In *CVPR*, 2011.

[9] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing Objects by their Attributes. In *CVPR*, 2009.

[10] Y. Fu, T. M. Hospedales, T. Xiang, J. Xiong, S. Gong, Y. Wang, and Y. Yao. Robust subjective visual property prediction from crowdsourced pairwise labels. *CoRR*, abs/1501.06202, 2015.

[11] W. Hsiao and K. Grauman. Learning the Latent Look: Unsupervised Discovery of a Style-Coherent Embedding from Fashion Images. In *ICCV*, 2017.

[12] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, 2007.

[13] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *PAMI*, 20(11):1254–1259, 1998.

[14] D. Jayaraman and K. Grauman. Zero-shot recognition with unreliable attributes. In *ECCV*, 2014.

[15] T. Joachims. Optimizing Search Engines Using Click-through Data. In *SIGKDD*, 2002.

[16] T. Judd, K. Ehinger, F. Durand, and A. Torralba. Learning to Predict Where Humans Look. In *ICCV*, 2009.

[17] S. Kazemzadeh, V. Ordonez, M. Matten, and T. L. Berg. Referit game: Referring to objects in photographs of natural scenes. In *EMNLP*, 2014.

[18] S. Kong, X. Shen, Z. Lin, R. Mech, and C. Fowlkes. Deep Understanding of Image Aesthetics. In *ECCV*, 2016.

[19] A. Kovashka and K. Grauman. Discovering attribute shades of meaning with the crowd. *IJCV*, 114(1):56–73, 2015.

[20] A. Kovashka, D. Parikh, and K. Grauman. Whittlesearch: Image search with relative attribute feedback. In *CVPR*, 2012.

[21] A. Kovashka, D. Parikh, and K. Grauman. WhittleSearch: Interactive Image Search with Relative Attribute Feedback. *IJCV*, 115(2):185–210, 2015.

[22] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012.

[23] G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg. Baby Talk: Understanding and Generating Image Descriptions. In *CVPR*, 2011.

[24] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Attribute and simile classifiers for face verification. In *ICCV*, 2009.

[25] C. Lampert, H. Nickisch, and S. Harmeling. Learning To Detect Unseen Object Classes by Between-Class Attribute Transfer. In *CVPR*, 2009.

[26] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang. Deep-Fashion: Powering Robust Clothes Recognition and Retrieval with Rich Annotations. In *CVPR*, 2016.

[27] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep Learning Face Attributes in the Wild. In *ICCV*, 2015.

[28] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[29] S. Maji and G. Shakhnarovich. Part and attribute discovery from relative annotations. *IJCV*, 108(1):82–96, 2014.

[30] J. Mao, J. Huang, A. Toshev, O. Camburu, A. Yuille, and K. Murphy. Generation and Comprehension of Unambiguous Object Descriptions. In *CVPR*, 2016.

[31] M. Mitchell, K. Deemter, and E. Reiter. Generating Expressions that Refer to Visible Objects. In *North American Chapter of the Asssociation for Computational Linguistics (NAACL)*, 2013.

[32] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *IJCV*, 42(3):145–175, 2001.

[33] J. Pan, E. Sayrol, X. Giro-i Nieto, K. McGuinness, and N. E. O'Connor. Shallow and deep convolutional networks for saliency prediction. In *CVPR*, 2016.

[34] D. Parikh and K. Grauman. Relative attributes. In *ICCV*, pages 502–510, 2011.

[35] G. Patterson and J. Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. *IJCV*, 108(1):59–81, 2014.

[36] J. C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, pages 61–74. MIT Press, 1999.

[37] A. Sadovnik, A. Gallagher, D. Parikh, and T. Chen. Spoken attributes: Mixing binary and relative attributes to say the right thing. In *ICCV*, 2013.

[38] R. Sandeep, Y. Verma, and C. Jawahar. Relative parts: Distinctive parts for learning relative attributes. In *CVPR*, 2014.

[39] B. Siddiquie, R. S. Feris, and L. S. Davis. Image Ranking and Retrieval based on Multi-Attribute Queries. In *CVPR*, 2011.

[40] K. Singh and Y. J. Lee. End-to-End Localization and Ranking for Relative Attributes. In *ECCV*, 2016.

[41] Y. Souri, E. Noury, and E. Adeli. Deep Relative Attributes. In *ACCV*, 2016.

[42] M. Spain and P. Perona. Measuring and Predicting Object Importance. *IJCV*, 91(1):59–76, 2011.

[43] N. Turakhia and D. Parikh. Attribute dominance: What pops out? In *ICCV*, 2013.

[44] X. Yan, J. Yang, K. Sohn, and H. Lee. Attribute2Image: Conditional Image Generation from Visual Attributes. In *ECCV*, 2016.

[45] X. Yang, T. Zhang, C. Xu, S. Yan, M. S. Hossain, and A. Ghoneim. Deep relative attributes. *IEEE Transactions on Multimedia*, 18(9):1832–1842, 2016.

[46] A. Yu and K. Grauman. Fine-Grained Visual Comparisons with Local Learning. In *CVPR*, 2014.

[47] A. Yu and K. Grauman. Just noticeable differences in visual attributes. In *ICCV*, 2015.

[48] A. Yu and K. Grauman. Semantic jitter: Dense supervision for visual comparisons via synthetic images. In *ICCV*, 2017.

[49] B. Zhao, J. Feng, X. Wu, and S. Yan. Memory-Augmented Attribute Manipulation Networks for Interactive Fashion Search. In *CVPR*, 2017.