Designing Category-Level Attributes for Discriminative Visual Recognition[*]

Felix X. Yu[†], Liangliang Cao[§], Rogerio S. Feris[§], John R. Smith[§], Shih-Fu Chang[†]
[†] Columbia University [§] IBM T. J. Watson Research Center
{yuxinnan, sfchang}@ee.columbia.edu {liangliang.cao, rsferis, jsmith}@us.ibm.edu

Abstract

Attribute-based representation has shown great promises for visual recognition due to its intuitive interpretation and cross-category generalization property. However, human efforts are usually involved in the attribute designing process, making the representation costly to obtain. In this paper, we propose a novel formulation to automatically design discriminative "category-level attributes", which can be efficiently encoded by a compact category-attribute matrix. The formulation allows us to achieve intuitive and critical design criteria (category-separability, learnability) in a principled way. The designed attributes can be used for tasks of cross-category knowledge transfer, achieving superior performance over well-known attribute dataset Animals with Attributes (AwA) and a large-scale ILSVRC2010 dataset (1.2M images). This approach also leads to state-of-the-art performance on the zero-shot learning task on AwA.

1. Introduction

Visual attributes have received renewed attention by the computer vision community in the past few years. The term "attribute" often refers to human nameable properties (*e.g.*, furry, striped, black) that are shared across categories, thereby enabling applications of leveraging knowledge learned from known categories to recognize novel categories, *a.k.a*, cross-category knowledge transfer. Such applications include recognizing unseen categories with no training examples, or *zero-shot* learning [13], and description of images containing unfamiliar objects [8]. "Attributes" have also been used to denote shareable properties of objects without concise semantic names (*e.g.*, dogs and cats have it but sharks and whales don't [8]) for improved discrimination.

The effectiveness of attribute-based representations has benefited broad applications, including face verification [12], image retrieval [25,34], action recognition [15], image-to-text generation [2], fine-grained visual categorization [6], and classification with humans-in-the-loop [4,20].

Problem. Designing attributes usually involves manually picking a set of words that are descriptive for the images under consideration, either heuristically [8] or through knowledge bases provided by domain specialists [13]. After deciding the set of attributes, additional human efforts are needed to label the attributes, in order to train attribute classifiers. The required human supervision hinders scaling up the process to develop a large number of attributes. More importantly, a manually defined set of attributes (and the corresponding attribute classifiers) may be intuitive but not discriminative for the visual recognition task.

Solution. In this paper, we propose a scalable approach of *automatically* designing *category-level attributes* for discriminative visual recognition. Our approach is motivated by [13,18], in which the attributes are defined by concise semantics, and then manually related to the categories as a *category-attribute matrix* (Figure 1). This matrix characterizes each category (row) in terms of the pre-defined attributes (columns). For example, polar bear is non-white, black, non-blue, *etc*. This matrix is critical for the subsequent process of category-level knowledge transfer.

Similar to characterizing categories as a list of attributes, attributes can also be expressed as how they relate to the known categories. For example, we can say the second attribute of Figure 1 characterizes the property that has high association of polar bear, and low association of walrus, lion, *etc*. Based on the above intuition, given the images with category labels (a multi-category dataset), we propose to automatically design a category-attribute matrix to *define* the attributes. Such attributes are termed as *category-level attributes*. The designed attributes will not have concise names as the manually specified attributes, but they can be loosely interpreted as relative associations of the known categories. Because multi-category datasets are widely available in the computer vision community, no additional hu-

[*] Supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D11PC20070. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBC, or the U.S. Government.

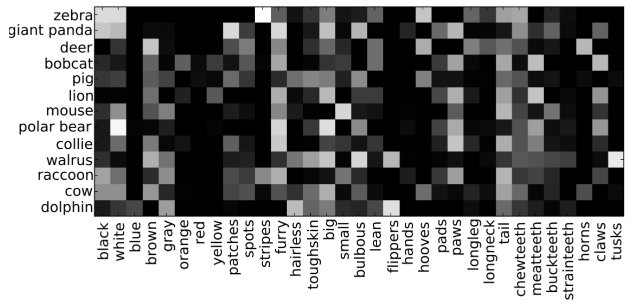

Figure 1. Manually defined category-attribute matrix copied from [13]: the rows are the categories, and the columns are the attributes. This matrix is obtained from human judgments on the "relative strength of association" between attributes and animal categories.

man efforts are needed in the above process.

Our work makes the following unique contributions:

- We propose a principled framework of using category-level attributes for visual recognition (Section 3.1).
- We theoretically demonstrate that discriminative category-level attributes should have the properties of category-separability and learnability (Section 3.2).
- Based on this analysis, an efficient algorithm is proposed for scalable design of attributes (Section 4).
- We conduct comprehensive experiments (Section 5) to demonstrate the effectiveness of our approach in recognizing known and novel categories. Our method achieves the state-of-the-art result on the zero-shot learning task.

2. Related Works

2.1. Designing Semantic Attributes

Traditionally, the semantic attributes are designed by manually picking a set of words that are descriptive for the images under consideration [8,13,16]. Similar way has also been explored for designing "concepts" in multimedia [17,31] and computer vision [14,26,27]. The concepts are sometimes manually or automatically organized into a hierarchical structure (ontology) to characterize different levels of semantics [17,26]. To alleviate the human burdens, [2] proposes to automatically discover attributes by mining the text and images on the web, and [23] explores the "semantic relatedness" through online knowledge source to relate the attributes to the categories. In order to incorporate discriminativeness for the semantic attributes, [6,19] propose to build nameable and discriminative attributes with human-in-the-loop. Compared to the above manually designed semantic attributes, our designed attributes cannot be used to describe images with concise semantic terms, and they may not capture subtle non-discriminative visual patterns of individual images. However, the category-level attributes can be automatically and efficiently designed for discriminative visual recognition, leading to effective solutions and even state-of-the-art performance on the tasks that traditionally achieved with semantic attributes.

2.2. Designing Data-Driven Attributes

Non-semantic "data-driven attributes" have been explored to complement semantic attributes with various forms. [12] combines semantic attributes with "simile classifiers" for face verification. [32] proposes data-driven "concepts" for event detection. [15] extends a set of manually specified attributes with data-driven attributes for improved action recognition. [24] extends a semantic attribute representation with extra non-interpretable dimensions for enhanced discrimination. [3,10,22] use the large-margin framework to model attributes for objective recognition. [7,30] use attribute-like latent models to improve object recognition. The highly efficient algorithm, and the unique capability of zero-shot learning, differentiate the proposed methodology from the above approaches.

The category-level attribute definition can be seen as a generalization of the discriminative attributes used in [8]. Instead of randomly generating the "category split" as in [8], we propose a principled way to *design* the category-level attributes.

3. A Learning Framework for Visual Recognition with Category-Level Attributes

3.1. The Framework

We propose a framework of using attributes as mid-level cues for multi-class classification on *known* categories. And the error of such classification scheme is used to measure the discriminativeness of attributes. Suppose there are k categories, and l attributes. The category-attribute matrix (definition of attributes) is denoted as $\mathbf{A} \in \mathbb{R}^{k \times l}$, in which the columns $\{\mathbf{A}_{\cdot i}\}_{i=1}^{l}$ define l category-level attributes, and the rows $\{\mathbf{A}_{i \cdot}\}_{i=1}^{k}$ correspond to k known categories.

Definition 1. *For an input image $\mathbf{x} \in \mathcal{X}$ (as low-level features), we define the following two steps to utilize attributes as mid-level cues to predict its category label $y \in \mathcal{Y}$.*
***Attribute Encoding**: Compute l attributes by attribute classifiers $\mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), ..., f_l(\mathbf{x})]^T$ in which $f_i(\mathbf{x}) \in \mathbb{R}$ models the strength of the i-th attribute for $\mathbf{x}$.*
***Category Decoding**: Choose the closest category (row of $\mathbf{A}$) in the attribute space (column space of $\mathbf{A}$):*

$$\arg\min_{i} \parallel \mathbf{A}_{i\cdot} - \mathbf{f}(\mathbf{x})^T \parallel . \qquad (1)$$

Because $\mathbf{A}$ is real valued, a unique solution for Equation 1 can be reached. Figure 2 illustrates using two attributes to discriminate cats and dogs.

Definition 2. *Designing discriminative category-level attributes is to find a category-attribute matrix $\mathbf{A}$, as well as the attribute classifiers $\mathbf{f}(\cdot)$ to minimize the multi-class classification error.*

Why the Framework. The above framework is motivated by two previous studies: learning attributes based on category-attribute matrix [13], and Error Correcting Output Code (ECOC) [1,5]. Multiple previous research can be unified into the framework by firstly setting $\mathbf{A}$ as a pre-defined matrix, and then modeling $\mathbf{f}(\cdot)$ accordingly. For example, in the previous studies, $\mathbf{A}$ was set as a manually defined matrix [13], a random matrix (discriminative attributes [8]), or a k-dimensional square matrix with diagonal elements as 1 and others as -1. The last case is exactly the one-vs-all approach, in which an attribute is equivalent to a single category. When applied for recognizing novel categories, such attributes are termed as category-level semantic features, or, *classemes* [27,31].

Unlike the manual attributes and classemes, the designed attributes are without concise semantics. However the category-level attributes are more intuitive than mid-level representations defined on low-level features, reviewed in Section 2.2. In fact, our attributes can be seen as *soft* groupings of categories, with analogy to the idea of building taxonomy or concept hierarchy in the library science. We provide a discussion on the semantic aspects of the proposed method in the supplementary technical report [33].

In addition, by defining attributes based on a set of known categories, we are able to develop a highly efficient algorithm to design the attributes (Section 4). It also enables a unique and efficient way for doing zero-shot learning (Section 5.3).

3.2. Theoretical Analysis

In this section, we theoretically show the properties of good attributes in a more explicit form. Specifically, we bound the empirical multi-class classification error in terms of attribute encoding error and a property of the category-attribute matrix, as illustrated in Figure 2.

Formally, given training examples $\{\mathbf{x}_i, y_i\}_{i=1}^m$, in which $\mathbf{x}_i \in \mathcal{X}$ is the feature, and $y_i \in \mathcal{Y}$ is the category label associated with $\mathbf{x}_i$:

Definition 3. *Define ϵ as the average encoding error of the attribute classifiers $\mathbf{f}(\cdot)$, with respect to the category-attribute matrix $\mathbf{A}$.*

$$\epsilon = \frac{1}{m} \sum_{i=1}^{m} \| \mathbf{A}_{y_i \cdot} - \mathbf{f}(\mathbf{x}_i) \|. \tag{2}$$

Definition 4. *Define ρ as the minimum row separation of the category-attribute matrix $\mathbf{A}$*

$$\rho = \min_{i \neq j} \| \mathbf{A}_{i \cdot} - \mathbf{A}_{j \cdot} \| . \tag{3}$$

Theorem 1. *The empirical error of multi-class classification is upper bounded by $2\epsilon/\rho$.*

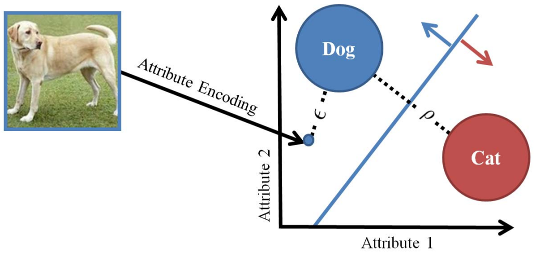

Figure 2. Discriminating dogs and cats, with two attributes. Each category (row of $\mathbf{A}$) is a template vector in the attribute space (column space of $\mathbf{A}$). ρ is the row separation of the category-attribute matrix. A new image of dog can be represented as an attribute vector through attribute encoding, and ϵ is the encoding error. In order for the image not to be mistakenly categorized as cat, we prefer smaller ϵ and larger ρ.

The proof of the theorem is provided in the supplementary technical report [33]. The message delivered by the bound is very intuitive. It tells us discriminative attributes should have the following properties, illustrated in Figure 2:

- **Category-separability.** We want ρ to be large, *i.e.* the categories should be separated in the attribute space.
- **Learnability.** We want ϵ to be small, meaning that the attributes should be learnable. This also implies that attributes should be shared across "similar" categories.

In addition, we also want the attributes to be non-redundant, otherwise we may get a large amount of identical attributes. In this paper, the redundancy is measured as

$$r = \frac{1}{l} \| \mathbf{A}^T \mathbf{A} - \mathbf{I} \|_F^2, \tag{4}$$

in which $\| \cdot \|_F$ is the Frobenius norm.

4. The Attribute Design Algorithm

Based on the above analysis, we propose an efficient and scalable algorithm to design the category-attribute matrix $\mathbf{A}$, and to learn the attribute classifiers $\mathbf{f}(\cdot)$. The algorithm is fully automatic given images with category labels.

4.1. Designing the Category-Attribute Matrix

To optimize the category-attribute matrix $\mathbf{A}$ (definition of attributes), we first consider the objective function in the following form, without the non-redundancy constraint:

$$\max_{\mathbf{A}} J(\mathbf{A}) = J_1(\mathbf{A}) + \lambda J_2(\mathbf{A}), \tag{5}$$

in which $J_1(\mathbf{A})$ induces separability (larger ρ), and $J_2(\mathbf{A})$ induces learnability (smaller ϵ). To benefit the algorithm, we set $J_1(\mathbf{A})$ as sum of all distances between every two rows of $\mathbf{A}$, encouraging every two categories to be separable in the attribute space.

$$J_1(\mathbf{A}) = \sum_{i,j} \| \mathbf{A}_{i \cdot} - \mathbf{A}_{j \cdot} \|_2^2 . \tag{6}$$

Algorithm 1 Designing the category-attribute matrix

```
Initialize $\mathbf{R} = \mathbf{Q}$, and $\mathbf{A}$ as an empty matrix, solve Equation 9 by sequentially learning $k$ additional columns.
for $i = 1 : k$ do
    Solve Equation 10 to get $\mathbf{a}$
    Add the new column $\mathbf{A} \leftarrow [\mathbf{A}, \mathbf{a}]$
    Update[1] $\mathbf{R} \leftarrow \mathbf{R} - \eta \mathbf{a}\mathbf{a}^T$
end for
```

We set $J_2(\mathbf{A})$ as a proximity preserving regularizer

$$J_2(\mathbf{A}) = -\sum_{i,j} S_{ij} \parallel \mathbf{A}_{i\cdot} - \mathbf{A}_{j\cdot} \parallel_2^2, \tag{7}$$

in which S_{ij} measures the category-level visual proximity between the category i and category j. The intuition is that if two categories are visually similar, we expect them to share more attributes, otherwise the attribute classifiers will be hard to learn. The construction of the visual proximity matrix $\mathbf{S} \in \mathbb{R}^{k \times k}$ will be presented in Section 4.3.

It is easy to show that

$$J(\mathbf{A}) = \text{Tr}(\mathbf{A}^T \mathbf{Q} \mathbf{A}), \quad \mathbf{Q} = \mathbf{P} - \lambda \mathbf{L}, \tag{8}$$

in which $\mathbf{P}$ is with diagonal elements being $k - 1$ and all the other elements -1, and $\mathbf{L}$ is the Laplacian of $\mathbf{S}$ [28].

Considering the non-redundant objective, if we force the designed attributes to be strictly orthogonal to each other, *i.e.* $\mathbf{A}^T \mathbf{A} = \mathbf{I}$, the problem can be solved efficiently by a single step, *i.e.* $\mathbf{A}$ combines the top eigenvectors of $\mathbf{Q}$. However, just like PCA, the orthogonal constraint will result in low-quality attributes, because of the fast decay of eigenvalues. So we relax the strict orthogonal constraint, and solve the following problem:

$$\max_{\mathbf{A}} \quad \text{Tr}(\mathbf{A}^T \mathbf{Q} \mathbf{A}) - \beta \parallel \mathbf{A}^T \mathbf{A} - \mathbf{I} \parallel_F^2 . \tag{9}$$

Without loss of generality, we require the columns of $\mathbf{A}$ (attributes) to be $l2$ normalized. We propose to incrementally learn the columns of $\mathbf{A}$. Specifically, given an initialized $\mathbf{A}$, optimizing an additional column $\mathbf{a}$ is to solve the following optimization.

$$\max_{\mathbf{a}} \quad \mathbf{a}^T \mathbf{R} \mathbf{a} \qquad \text{s.t.} \quad \mathbf{a}^T \mathbf{a} = 1, \tag{10}$$

in which $\mathbf{R} = \mathbf{Q} - \eta \mathbf{A}\mathbf{A}^T$, $\eta = 2\beta$. This is a Rayleigh quotient problem, with the optimal $\mathbf{a}$ as the eigenvector of $\mathbf{R}$ with the largest eigenvalue. The overall algorithm is described in Algorithm 1. The algorithm greedily finds additional non-redundant attributes, with desired properties.

4.2. Learning the Attribute Classifiers

After getting the real-valued category-attribute matrix $\mathbf{A}$, the next step is to learn the attribute classifiers $\mathbf{f}(\cdot)$. We assume each classifier $\{f_i(\cdot)\}_{i=1}^l$ can be learned independently. Specifically, suppose $f_i(\cdot)$ can be represented by a linear model $\mathbf{w}_i$, our solution is to solve a large-margin classification problem with weighted slack variables.

$$\begin{aligned} \min_{\mathbf{w}_i, \xi} \quad & \parallel \mathbf{w}_i \parallel_2^2 + C \sum_{j=1}^{m} |A_{y_j,i}| \xi_j \\ \text{s.t.} \quad & \text{sign}(A_{y_j,i}) \mathbf{w}_i^T \mathbf{x}_j \geq 1 - \xi_j \\ & \xi_j \geq 0, \quad j = 1...m \end{aligned} \tag{11}$$

in which the binarized category-attribute matrix element sign$(A_{y_j,i})$ defines the presence/non-presence of the ith attribute for $\mathbf{x}_j$. The idea is to put higher penalties for misclassified instances from categories with stronger category-attribute association. Generalizing to kernel version is straightforward.

4.3. Building the Visual Proximity Matrix S

In the proposed algorithm in Section 4.1, one important issue is to build the visual proximity matrix $\mathbf{S} \in \mathbb{R}^{k \times k}$ used in Equation 7. This matrix is key towards making the attributes learnable, and sharable across categories. Similar to [28], we first build a distance matrix $\mathbf{D} \in \mathbb{R}^{k \times k}$, in which D_{ij} measures the distance between category i and j. $\mathbf{S}$ is modeled as a sparse affinity matrix, with the non-zero elements $S_{ij} = e^{-D_{ij}/\sigma}$.

$\mathbf{D}$ is built dependent on *type* of kernel used for learning the attribute classifiers $\mathbf{f}(\cdot)$ (Section 4.2)[2]. When nonlinear kernels are used, SVM margins, of $k(k-1)/2$ *one-vs-one* SVMs modeled on low-level features are used as distance measurement for categories; when linear kernels are used (which is usually used for large-scale problems), we simply use the distances of category centers (category mean of the low-level features) as distance measurements. Because the category centers can be pre-computed, the latter process is very fast, with computational complexity linear to # images, and quadratic to # categories.

4.4. Discussions

Efficiency and Scalability. The attribute design algorithm requires no expensive iterations on the image features. The computational complexity of designing an attribute (a column of $\mathbf{A}$) is as efficient as finding the eigenvector with the largest eigenvalue of matrix $\mathbf{R}$ in Equation 10 (quadratic to # categories). For example, on a 6-core Intel 2.5 GHz

[1] Because $\mathbf{A}\mathbf{A}^T = \sum_i A_i A_i^T$, in each iteration $\mathbf{R}$ can be updated as $\mathbf{R} \leftarrow \mathbf{R} - \eta \mathbf{a}\mathbf{a}^T = \mathbf{Q} - \eta \mathbf{A}\mathbf{A}^T$ for efficiency.

[2] The visual proximity matrix $\mathbf{S}$ is only dependent on the kernel type, not the learned attribute classifiers. Therefore, designing attributes (Section 4.1) and learning the attribute classifiers (Section 4.2) are two sequential steps, requiring no expensive iterations on the image features.

workstation, it just takes about 1 hour to design 2,000 attributes based on 950 categories on the large-scale ILSVRC2010 dataset (Section 5.2).

Known Categories vs. Novel Categories. Though the above algorithm is for designing attributes to discriminate known categories, the application of the designed attributes is for recognizing *novel* categories, the categories that are not used in the attribute designing process. In specific, we will show by experiment:

- The designed attributes are discriminative for novel, yet related categories (Section 5.2 AwA dataset).
- The designed attributes are discriminative for general novel categories, provided we can design a large amount of attributes based on a diverse set of known categories (Section 5.2 ILSVRC2010 dataset).
- The attributes are effective for the task of zero-shot learning (Section 5.3).

5. Experiments

Datasets. We evaluate the performance of the designed attributes on Animal with Attributes (AwA) [13], and ILSVRC2010 datasets[3]. AwA contains 30,475 images of 50 animal categories. Associated with the images, there is a manually designed category-attribute matrix of 85 attributes shown in Figure 1. ILSVRC2010 contains 1.2M images from 1,000 diverse categories. The experiments are performed 10 times, and we report the mean performance.

Baselines. We first demonstrate that our designed category-level attributes are more discriminative than other *category-level representations*. In the task of discriminating known categories (Section 5.1), we use the framework proposed in Section 3, and compare the performance of the designed attributes with the manual attributes [13] (85 manually defined attributes with a manually specified category-attribute matrix), random category-level attributes [8] (attributes defined as a randomly generated category-attribute matrix), and one-vs-all classifiers (equivalent to attributes defined as an matrix, with diagonal elements as 1 and others as -1). In the task of novel category recognition in Section 5.2, we use the extracted attributes as features to perform classification on the images of novel categories. Our approach is compared with the manual attributes, random attributes, classemes [27] (one-vs-all classifiers learned on the known categories), and low-level features (one-vs-all classification scheme based on low-level features of the novel categories). We also test the retrieval and classification performance of our approaches based on the large-scale ILSVRC2010 data.

To demonstrate the capability of zero-shot learning of the designed attributes, we compare our approach with the best published results to date in Section 5.3.

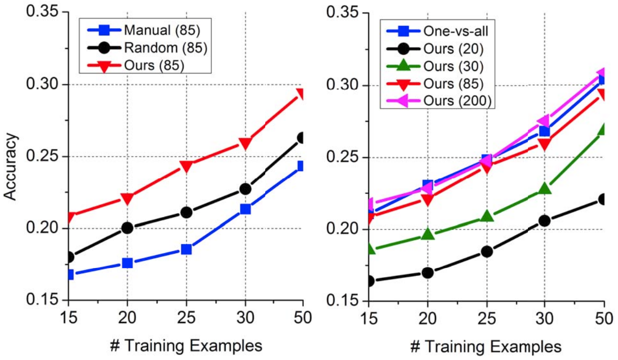

Figure 3. Multi-class classification accuracy on known categories. The numbers in bracket are # attributes. The standard deviation is around 1%.

Measurement	Designed	Manual	Random
Encoding error ϵ	**0.03**	0.07	0.04
Minimum row separation ρ	**1.37**	0.57	1.15
Average row separation	**1.42**	1.16	1.41
Redundancy r	**0.55**	2.93	0.73

Table 1. Properties of different attributes. The number of attributes is fixed as 85. Encoding error ϵ is defined in Equation 2. Minimum row separation ρ is defined in Equation 3. Averaged row separation is value of the objective function in Equation 6. Redundancy r is defined in Equation 4. The category-attribute matrices are column-wise $l2$ normalized in order to be comparable. The measurements are computed on the test set.

5.1. Discriminating Known Categories

In this section, we verify the multi-class classification performance and other properties described in Section 3.2 on 40 *known* categories of AwA dataset. The attributes are designed based on 40 training categories defined in [13]. The same low-level features (10,940D), and kernel (χ^2 with bandwidth as 0.2 times median distance) are used. For random attributes, each element of the random category-attribute matrix is generated uniformly from $[-1, 1]$[4].

We select different amount of images per category for training, 25 images per category for testing, and 10 images per category for validation. The parameters are tuned based on the validation set. The margins of $40 \times 39/2 = 780$ one-vs-one classifiers on the training data are used as distance measurements $\mathbf{D}$ of animal categories (the C parameter for one-vs-one SVMs is simply fixed as 10). The visual proximity matrix $\mathbf{S}$ is built as the mutual 10-NN adjacent matrix with bandwidth parameter σ as 0.5 times the average distance [28]. We first fix the weighed SVM penalty $C = 2$, and tune $\lambda \in \{2, 3, 4, 5\}$, $\eta \in \{6, 8, 10, 12, 14\}$. Then we tune $C \in \{0.02, 0.2, 2, 20\}$.

Figure 3 demonstrates the performance of multi-class

[3] http://www.image-net.org/challenges/LSVRC/2010/download-public

[4] Other alternatives including binary random matrix, sparse binary random matrix, yield similar performance.

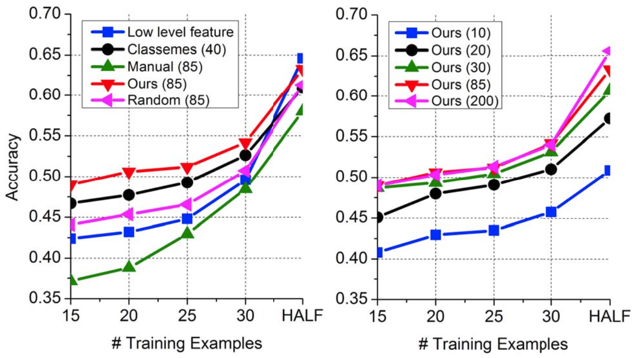

Figure 4. Multi-class classification accuracy on novel categories. The 64.6% accuracy with one-vs-all classifier using 50/50 (HALF) split is similar to the performance (65.9%) reported in [13]. The numbers in bracket are # attributes. The standard deviation is around 1%.

classification. Table 1 further verifies the properties of the designed attributes.

- The designed attributes perform significantly better than the manual attributes and random category-level attributes (Figure 3 left).
- The designed attributes is competitive to, if not better than the low-level features paired with one-vs-all χ^2 classifiers (Figure 3 right). The designed attributes significantly outperform the the one-vs-all classifiers (known as classemes) for the task of recognizing novel categories (Section 5.2), due to the fact that classemes are not shared across categories.
- The designed attributes have smaller encoding error, larger row separation and smaller redundancy. This justifies the theoretical analysis in Section 3.2.

One interesting observation is that even the random category-attribute matrix has better properties compared to the manually defined category-attribute matrix (Table 1). The random attributes therefore outperform the manual attributes (Figure 3 left).

5.2. Discriminating Novel Categories

We show that the designed attributes are also discriminative for *novel* categories. Specifically, we use the attributes, and other kinds of category-level representations as *features*, to perform the multi-class classification (AwA, ILSVRC2010) and the category-level retrieval (ILSVRC2010) tasks.

Animals with Attributes. We use the 40 animal categories in Section 5.1 to design the attributes. Efficient linear SVM classifiers are trained based on different kinds of attribute features to perform classification on the images of 10 novel categories. The optimally tuned parameters in Section 5.1 are used for the task (The C parameter of linear SVM is tuned based on the same grid).

Figure 4 shows the performance. The designed attributes perform significantly better than other types of representations, especially with few training examples. This means that attributes are discriminative representation for the novel categories, by leveraging knowledge learned from known categories. As the # training images increases, the performance of low-level features is improved, due to sufficient supervision. Note that the manual attributes and classemes are with fixed dimensions, not extendable due to definition, whereas the dimension of the designed category-level attributes is scalable.

Method	Precision@50
Low-level feature	33.40
Classeme (950)	39.24
Ours (500)	39.85
Ours (950)	42.16
Ours (2,000)	**43.10**

Table 2. Category-level image retrieval result on 50 classes from ILSVRC2010. The numbers in bracket are # attributes. We closely follow the settings of [9].

	Percentage for training				
Method	1%	5%	10%	50%	100%
Low-level feature	35.55	52.21	57.11	66.21	**69.16**
Classemes (950)	38.54	51.49	56.18	64.31	66.77
Ours (500)	39.01	52.86	56.54	62.38	63.86
Ours (950)	41.60	55.32	59.09	65.15	66.74
Ours (2,000)	**43.39**	**56.51**	**60.36**	**66.91**	68.17

Table 3. Image classification accuracy on 50 classes from ILSVRC2010. The training set contains 54,636 images. The numbers in bracket are # attributes. Standard deviation is around 1%. We closely follow the settings of [9].

ILSVRC2010. In the previous experiments on AwA, we show that the designed attributes are discriminative for novel, yet related categories. We now demonstrate that the designed attributes can be discriminative for general novel categories, provided that we can design a large amount of attributes based on a diverse set of known categories. The ILSVRC2010 dataset is used for this experiment. Following the settings in [9], the low-level features are 4,096 dimensional fisher vectors [21]. 950 categories are used as known categories to design attributes. We test the performance of using attribute features for category-level retrieval and classification on the remaining 50 disjoint categories.

The distances of category centers (based on low-level features) are used as distance measurements $\mathbf{D}$ of categories. The visual proximity matrix $\mathbf{S}$ is built as a 30-NN mutual adjacent matrix, with bandwidth parameter σ as 0.5 times the mean distance [28]. The attributes are trained by linear weighted SVM models. All other detailed experiment settings, including data splits, and ways for parameter tuning are identical to [9].

We first test the performance of the designed attributes for category-level image retrieval. 1,000 randomly selected images are used as queries to retrieve the nearest neighbors from the remaining 67,295 images. Table 2 shows the performance in terms of precision@50. The designed attributes outperform low-level features and classemes, even with 500 dimensions. And 2,000-dimensional attributes outperform the baselines by 9.70% and 3.86% respectively.

Next, we use the attribute feature, combined with linear SVM classifiers to classify the images of the 50 novel categories. 80% of the data are used for training (54,636 images), 10% for testing, and 10% for validation. Table 3 shows multi-class classification accuracy, using different amount of training images from the training set. Similar to the experiments on AwA dataset, attribute representation outperforms the baselines, especially when training with small amount of examples. It means attributes are effective for leveraging information of the known categories to recognize novel categories. As the amount of training images increases, the performance of low-level features goes up, due to sufficient amount of supervisions.

5.3. Zero-Shot Learning

Building the New Category-Attribute Matrix. Zero-shot learning can be seen as an special case of recognizing novel categories, without training data. In such case, human knowledge [13,18] is required to build a new category-attribute matrix $\tilde{\mathbf{A}} \in \mathbb{R}^{p\times l}$, to relate the p novel categories to the l designed attributes. After that, we can follow the framework in Section 3.1 to recognize the novel categories. However, for each designed attribute in our approach, there is no guarantee that it possesses a coherent human interpretation. For example, while some may say the visual property separating tiger and zebra from cat and dog is "striped", others may say it is the sizes of animals that matter. Therefore, given a new animal, *e.g.* skunk (both striped and small), the humans may come up with different answers.

Motivated by the fact that the visual proximity matrix $\mathbf{S}$ in Equation 7 is central to the attribute design process, we propose a fairly straightforward solution: similar to [29], given each novel category, and k known categories, we ask the user to find the top-M visually similar categories. The user is free to use any similarity interpretation they wish. We will then have a similarity matrix $\tilde{\mathbf{S}} \in \{0,1\}^{p\times k}$, in which $\tilde{S}_{ij}$ is the binary similarity of the i-th novel category and the j-th known category. The novel categories are related to the designed attributes by the simple weighted sum:

$$\tilde{\mathbf{A}} = \tilde{\mathbf{S}}\mathbf{A} \tag{12}$$

The amount of human interaction is minimal for the above approach, independent on the number of attributes.

Experiment Results. We test the zero-shot learning performance on the AwA dataset, with same settings of [13] (40 animal categories for training and 10 categories for testing). For each novel category, we ask the users to provide up to top-5 similar categories when building the similarity matrix. Empirically, fewer categories cannot fully characterize the visual appearance of the animals, and more categories will lead to more human burdens. Ten graduate students, who were not aware of the zero-shot learning experiments, were included in the study. When performing the the tasks, they were asked to think about visual similarities, rather than similarities otherwise. The time spent for the task ranges from 15 to 25 minutes. Because there is no validation set for zero-shot learning, we empirically set λ, η and SVM penalty C as 3, 15 and 20, throughout the experiments. The performance is not sensitive to the parameters for the range described in Section 5.1.

Method	# Attributes	Accuracy
Lampert *et al.* [13]	85	40.5
Yu and Aloimonos [35]	85	40.0
Rohrbach *et al.* [23]	-	35.7
Kankuekul *et al.* [11]	-	32.7
Ours	10	40.52 ± 4.58
Ours	85	42.27 ± 3.02
Ours	200	42.83 ± 2.92
Ours (Fusion)	200	**46.94**
Ours (Adaptive)	200	45.16 ± 2.75
Ours (Fusion + Adaptive)	200	**48.30**

Table 4. Zero-shot multi-class classification accuracy with standard deviation on the 10 novel animals categories.

Figure 4 shows the experiment results compared to various published baselines. Our approach achieves the state-of-the-art performance, even with just 10 attributes. The accuracy and robustness can be improved by using more attributes, and by averaging the multiple binary visual similarity matrices (Fusion). The former helps to fully explore the visual similarities $\tilde{\mathbf{S}}$, and the later helps to filter out noise from different users. We have achieved accuracy of 46.94%, which significantly outperforms all published results.

Adaptive Attribute Design. In the experiments above, the attributes are designed to be discriminative for the known categorizes. As a refinement for zero-shot learning, we can modify the algorithm to design attributes *adaptively* for discriminating the novel categories. This can be achieved by changing the first objective $J_1(\mathbf{A})$ (Section 4.1) to

$$\tilde{J}_1(\mathbf{A}) = J_1(\tilde{\mathbf{S}}\mathbf{A}). \tag{13}$$

In other words, we want to design a category-attribute matrix $\mathbf{A}$ which is specifically discriminative for the *novel* categories. The modified problem can be solved with minor modifications of the algorithm.

The last two rows of Table 4 demonstrate the performance of adaptive attribute design. Combined with averaged similarity matrix (Fusion + Adaptive), we have achieved multi-class classification accuracy of 48.30%, outperforming all published results with larger margin. The drawback for the adaptive attribute design is that we need to redesign the attributes for different tasks. Because the proposed attribute design algorithm is highly efficient, the drawback can be alleviated.

6. Conclusion

We propose a novel method for designing category-level attributes. Such attributes can be effectively used for tasks of cross-category knowledge transfer. Our future work is to incorporate concise semantics in the attributes, with the help of human interactions.

Acknowledgment We thank Yunchao Gong and Florent Perronnin for sharing the ILSVRC2010 features. We also thank Dong Liu, Michele Merler, Ming-Hen Tsai, and anonymous reviewers for their insightful suggestions.

References

[1] E. Allwein, R. Schapire, and Y. Singer. Reducing multiclass to binary: A unifying approach for margin classifiers. *JMLR*, 1:113–141, 2001.

[2] T. Berg, A. Berg, and J. Shih. Automatic attribute discovery and characterization from noisy web data. In *ECCV*, 2010.

[3] A. Bergamo, L. Torresani, and A. Fitzgibbon. Picodes: Learning a compact code for novel-category recognition. In *NIPS*, 2011.

[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] K. Crammer and Y. Singer. On the learnability and design of output codes for multiclass problems. *Machine Learning*, 47(2):201–233, 2002.

[6] K. Duan, D. Parikh, D. Crandall, and K. Grauman. Discovering localized attributes for fine-grained recognition. In *CVPR*, 2012.

[7] A. Farhadi, M. Tabrizi, I. Endres, and D. Forsyth. A latent model of discriminative aspect. In *CVPR*, 2009.

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

[9] Y. Gong, S. Lazebnik, A. Gordo, and F. Perronnin. Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. *PAMI*, PP(99):1, 2012.

[10] A. Gordoa, J. Rodrıguez-Serranoa, F. Perronnina, and E. Valvenyb. Leveraging category-level labels for instance-level image retrieval. In *CVPR*, 2012.

[11] P. Kankuekul, A. Kawewong, S. Tangruamsub, and O. Hasegawa. Online incremental attribute-based zero-shot learning. In *CVPR*, 2012.

[12] N. Kumar, A. Berg, P. Belhumeur, and S. Nayar. Attribute and simile classifiers for face verification. In *CVPR*, 2009.

[13] C. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*, 2009.

[14] L. Li, H. Su, E. Xing, and L. Fei-Fei. Object bank: A high-level image representation for scene classification and semantic feature sparsification. In *NIPS*, 2010.

[15] J. Liu, B. Kuipers, and S. Savarese. Recognizing human actions by attributes. In *CVPR*, 2011.

[16] D. Mahajan, S. Sellamanickam, and V. Nair. A joint learning framework for attribute models and object descriptions. In *ICCV*, 2011.

[17] M. Naphade, J. Smith, J. Tesic, S.-F. Chang, W. Hsu, L. Kennedy, A. Hauptmann, and C. J. Large-scale concept ontology for multimedia. In *IEEE Multimedia*, 2006.

[18] M. Palatucci, D. Pomerleau, G. Hinton, and T. Mitchell. Zero-shot learning with semantic output codes. In *NIPS*, 2009.

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

[20] A. Parkash and D. Parikh. Attributes for classifier feedback. In *ECCV*, 2012.

[21] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *ECCV*, 2010.

[22] M. Rastegari, A. Farhadi, and D. Forsyth. Attribute discovery via predictable discriminative binary codes. In *ECCV*, 2012.

[23] M. Rohrbach, M. Stark, G. Szarvas, I. Gurevych, and B. Schiele. What helps where–and why? semantic relatedness for knowledge transfer. In *CVPR*, 2010.

[24] V. Sharmanska, N. Quadrianto, and C. Lampert. Augmented attribute representations. In *ECCV*, 2010.

[25] B. Siddiquie, R. Feris, and L. Davis. Image ranking and retrieval based on multi-attribute queries. In *CVPR*, 2011.

[26] L. Torresani and A. Bergamo. Meta-class features for large-scale object categorization on a budget. In *CVPR*, 2012.

[27] L. Torresani, M. Szummer, and A. Fitzgibbon. Efficient object category recognition using classemes. In *ECCV*, 2010.

[28] U. Von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

[29] G. Wang, D. Forsyth, and D. Hoiem. Comparative object similarity for improved recognition with few or no examples. In *CVPR*, 2010.

[30] Y. Wang and G. Mori. A discriminative latent model of object classes and attributes. In *ECCV*, 2010.

[31] A. Yanagawa, S.-F. Chang, L. Kennedy, and W. Hsu. Columbia University's baseline detectors for 374 LSCOM semantic visual concepts. *Columbia University ADVENT Technical Report # 222-2006-8*, 2007.

[32] Y. Yang and M. Shah. Complex events detection using data-driven concepts. *ECCV*, 2012.

[33] F. Yu, L. Cao, R. Feris, J. Smith, and S.-F. Chang. Additional remarks on designing category-level attributes for discriminative visual recognition. *Columbia University Computer Science Department Technical Report # CUCS 007-13*, 2013.

[34] F. Yu, R. Ji, M.-H. Tsai, G. Ye, and S.-F. Chang. Weak attributes for large-scale image retrieval. In *CVPR*, 2012.

[35] X. Yu and Y. Aloimonos. Attribute-based transfer learning for object categorization with zero/one training example. In *ECCV*, 2010.