Journal of Naval Science and Engineering
2016, Vol. 12, No.1, pp.51-79
Bilgisayar Mühendisliği/Computer Engineering

UNSUPERVISED FEATURE LEARNING FOR MID-LEVEL DATA REPRESENTATION

Emrah ERGÜL[1], Mehmet KARAYEL[2], Oğuzhan TİMUŞ[3], Erkan KIYAK[4]

[1]*Dr.,Turkish Navy Inventory Control Center, Golcuk, Kocaeli, emergul13@yahoo.com*
[2]*Turkish Navy Inventory Control Center, Golcuk, Kocaeli, m.karayel.se@gmail.com*
[3]*Dr., Turkish Navy Inventory Control Center, Golcuk, Kocaeli, oguzhantimus@gmail.com.tr*
[4]*Dr., Turkish Naval Academy, Tuzla, İstanbul, ekiyak@dho.edu.tr*

Date of Receive: 20.11.2015 **Date of Acceptance: 16.02.2016**

ABSTRACT

Attribute based approaches are commonly used in recent years instead of low level features for image classification which is one of the most important problems in the field of computer vision. The most important advantage of attribute based approach is that learning can be performed similar to human by using attributes which makes sense for people. In this study, unsupervised attributes are developed in order to avoid human related problems in supervised attribute learning. In our proposed work, the attributes are generated as random binary and relative definitions. The process of random attribute generation simplifies the data modeling when compared to other work in the literature. In addition, a major problem which is the increasing the numbers of attributes in attribute based approaches is eliminated owing to the increasing the numbers of attributes easily. Furthermore, attributes are selected more wisely using simple applicable algorithm to improve the discriminative capacity of randomly generated attribute set for image classification. The proposed approaches are evaluated with the other similar attribute based studies comparatively in the literature based on the same data set (OSR-Open Scene Recognition). Experiments show that noteworthy performance increase is achieved.

ORTA SEVİYE VERİ TEMSİLİNDE DENETİMSİZ NİTELİK ÖĞRENİMİ

ÖZ

Bilgisayarla görme alanındaki en önemli problemlerden birisi olan imge sınıflandırma için öznitelik tabanlı klasik yaklaşımların yanı sıra nitelik tabanlı yaklaşımlar son yıllarda sıklıkla kullanılmaya başlanmıştır. Nitelik tabanlı yaklaşımların en önemli avantajı, insanlar için anlam ifade eden niteliklerin kullanılması vasıtasıyla insanoğluna benzer bir öğrenme yapılabilmesidir. Bu çalışmada, denetimli nitelik öğrenme sürecinde insan faktörü sebebiyle oluşabilecek sorunlardan kaçınmak amacıyla denetimsiz yaklaşım geliştirilmiştir. Denetimsiz yaklaşımımızda niteliklerin ikili ve göreceli olarak rastgele üretilmesi sayesinde nitelik öğrenme süreci, literatürdeki diğer denetimli ve denetimsiz yaklaşımlara göre daha kolay hale gelmiştir. Ayrıca, nitelik sayısının basit bir şekilde artırılması ile nitelik tabanlı yaklaşımlarda büyük bir problem olan nitelik sayısının artırılması basitleştirilmiştir. Rastgele üretilen nitelik kümesinin imge sınıflandırma için ayırt etme kapasitesini artırmak maksadıyla, rastgele üretilen nitelikler arasından en iyileri kolay uygulanabilir bir algoritma sayesinde seçilmiştir. Çalışmada önerilen yaklaşımlar literatürdeki diğer benzer nitelik tabanlı çalışmalarla aynı veri kümesi (OSR-Açık Alan Tanıma - Open Scene Recognition) üzerinden ve farklı sınıflandırıcılar kullanılarak test edilmiştir. Yapılan deneylerde denetimsiz öğrenilen göreceli niteliklerin dikkate değer bir performans artışı sağladığı görülmüştür.

Anahtar Kelimeler: Göreceli Nitelikler; Denetimsiz Nitelik Çıkartımı; Nitelik Seçimi; Görsel Tanıma.

Keywords: Relative Attributes; Unsupervised Feature Extraction; Attribute Selection; Visual Recognition.

1. INTRODUCTION

Attributes constitute intermediate layer data representation between the low-level image features (i.e. color/edge histograms, bag of visual words, quantized pixel values, GIST, SIFT, Fourier/Laplace/Hough/Wavelet transforms etc.) and the top level categories. Because attributes are common properties of the object categories, intermediate representations can be achieved by using classes in combinations with respect to the shared attributes, and this leads to generating new discriminative spaces for visual recognition.

Visual attributes are important for understanding object appearance and can be used for describing objects. In detail, visual attributes include color, modal, textural, functional, structural, and conceptual or any kind of semantic properties of objects. In addition to visual or semantic distinction, the representation of attribute is also varied as binary or relative. The presence or absence of an attribute in binary and the strength of an attribute in relative become important in attribute representation. One may think binary correlations (i.e. existence or absence of an attribute in a class) would be sufficient while the others claim real-valued ranking scores are essential to measure the attribute strength among categories [6,15].

Attributes can be learned by supervised or unsupervised manners. Supervised methods are firstly proposed in the literature and then unsupervised approaches become more popular. In supervised attribute learning, images are labeled with attributes by human effort. Hence, many difficulties occur. These difficulties can be summarized as; more general and intuitive attributes are determined instead of discriminative attributes which are indeed appropriate for classification purposes. In addition, some discriminative attributes may be overlooked or could not be expressed by words. Furthermore, erroneous attribute tagging can be performed. Finally, the process of attribute extraction become exhaustive and it takes a long time in large datasets that may contain many attributes [16]. In addition to above mentioned difficulties; attribute labeling of datasets in supervised methods needs a great deal of human laboring and budget. Moreover, extracting attributes by searching the related images on the internet as in [9] seems to be a clever idea for cost reduction, but the discovered attributes can be irrelevant with the image categories.

Since attributes are commonly shared amongst different top level categories, one of the major advantages of attributes is that fewer training examples are required to train an attribute and a classifier established on the basis of attributes. Consequently, the main idea is to learn attributes at the intermediate level for separating visual categories efficiently in attribute learning. However, the uppermost main target is to discriminate classes and it is not to learn some attributes perfectly.

In this work, we aim the image classification with the visual attributes which are used as the new feature space at mid-level. This kind of representation is achieved in an unsupervised way such that binary and relative attributes are learned by random binary predicates or class based relative orderings. Additionally, we select some of randomly generated orderings distinctively by implementing Kendall Tau metric which computes the distance between two sequences.

The contribution of this work is two folded. Firstly, we get unsupervised data representation at a new mid-level feature space with binary/relative attributes. The class based attributes are generated randomly, and binary SVM scores are used out of binary attributes while ordering scores are handled for relative ones. So the new feature space is assumablely expanded and established more discriminatively. On the other hand, we train the basis vectors of the new feature space with a very limited number of training instances. Secondly, we also select some of randomly generated attributes with a distance based algorithm where more discriminative sequences are picked. We also try three classifier (kNN, decision tree and SVM) for accuracy performance comparisons.

In Section 2, the development of attribute notion and attribute based approaches are explained in mixed form. In Section 3, our proposed algorithms based on random binary and relative attributes are introduced while in Section

4 experimental results are detailed. Finally, the experimental results are concluded.

2. RELATED WORK

The literature of attribute-based computer vision problems can be generally summarized in the types and extraction methods of attributes, applications and datasets on which they are implemented, and performance criteria in the experiments, as shown in Figure 1.

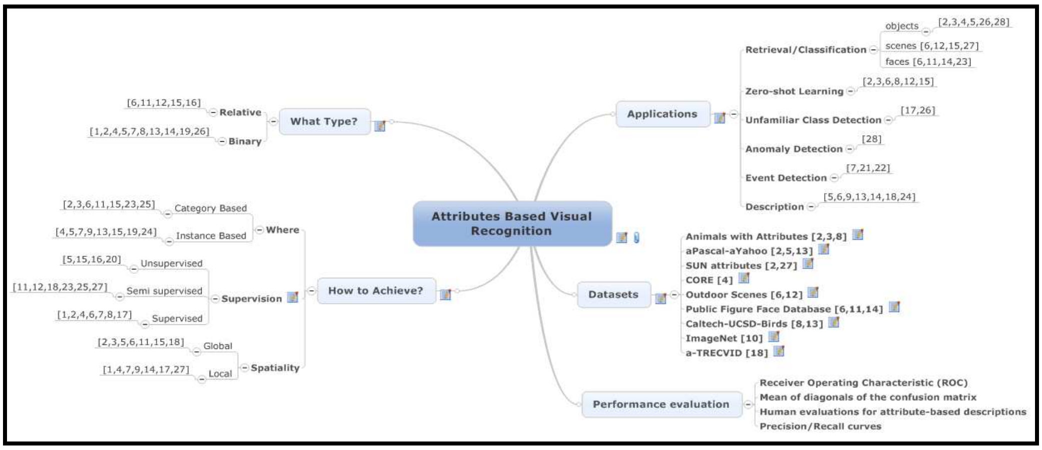

Figure 1: Summary of the literature works on attribute-based computer vision problems.

Ferrari and Zisserman propose a probabilistic generative model which infers whether an image contains a learned binary attribute and determines which regions over an object image the attributes may cover [1] in a weakly supervised manner. They use simple attribute like shape, color or texture; and two adjacent segments produce a complex attribute like spotted, striped, checked etc. Lampert et. al. study object recognition for categories which are not seen during training at all [2, 3].

Farhadi et. al. describe objects by the spatial arrangement of their appearance and part based attributes, and they benefit from the interactions between attributes [4]. In addition, Farhadi et. al. [5] embed object recognition problem into describing objects and mainly focus on attribute learning on the basis of semantic (i.e. nameable) attributes like object parts and discriminative attributes which are achieved by splitting the visual feature space into different regions by comparing some classes randomly in a binary fashion. The discriminative intuition is that some attributes cannot be nameable although they may be very useful for discrimination.

The relative attributes are first introduced by Parikh and Grauman in [6] with the assumption that semantically more enriched data descriptions and discrimination will be achieved if we use relative class memberships on attributes, instead of binary relations. SVM-like algorithm is implemented as a ranking function in which not only maximizing the margins between class boundaries but also ordering of classes over attributes space is aimed by using Newton’s method. They use predetermined nameable attributes and class orderings are given on these attributes in a supervised manner. The main restriction of the algorithm is that equality acceptance in category ordering according to an attribute. In the notion of equality acceptance, human can not differentiate two image belong to different categories. On the basis of the supervised relative attributes [8], many studies [33,34] whose aim is to learn more robust and precise relative attributes have emerged in the literature. In [33], it is claimed that relative attribute learning method is insufficient in indistinguishable image pairs which the human can’t sort or differentiate two image belong to different categories on the basis of an attribute. Namely, it is aimed to sort the image pairs which are assumed as equal situation in [6]. For this purpose, Bayesian local learning is proposed in [18]. In addition, instead of creating new method for generating relative attributes, Verma et al [34] improve the performance of the basic relative attribute method using patch-based features instead of global (GIST and Color Histogram) which are used in [6]. Verma et al claimed that their representation capture local shape in an image comparison to global features.

Simple solution to a multi-attribute query is to train a classifier for each attribute independently and combine their scores in retrieval. But some attribute conjunctions may be more useful since such combinations can be learned more easily and they discriminate visual data more. At this point, it is critical to determine which combinations of attributes should be trained without trying all combinations intensively. Rastegari et. al. [13] focus on learning more discriminative attributes by merging some of them, instead of learning each attribute individually. Kumar et. al. [14] open an interesting discussion about attributes in that similarity of faces with respect to other specific people as references may help for achieving more discriminative attributes for face verification, called ‘similes’. With such visual traits, for example, a face might be described as having a forehead like Barack Obama’s and eyes like Jennifer Lopez’s.

It is often intractable for a human to predefine and label all the attributes in large datasets explicitly. Furthermore, some attributes may be more valuable for recognition although we can’t name them. Ma et. al. [15] implement an algorithm to learn class-level relative attributes in an unsupervised manner, unlike [6] where relations between pairs of classes on attributes are already given relatively.

Instead of using pre-determined binary/relative attribute labels or class orderings on attributes, Karayel and Arica [16] follow the similar way of unsupervised attribute learning like in [5, 14, 15]. But in here, binary and relative attributes are produced completely randomly where classes are separated into positive and negative sides for binary attributes, while class ranks are selected for each relative attributes. Wang and Mori [17] purpose to model objects discriminatively for classification as the dependencies among attributes are captured using an undirected graphical model built from a training set. The main distinction from other works is that they unify object class and attribute predictions in a joint framework since classes and their attributes are closely related concepts.

So far, we mainly build category-attribute correlation matrices or dependency matrix among attributes for object recognition. Assuming that we have a limited number of nameable attributes which are pre-determined, such matrices would not be sufficient for large-scale computer vision problems. Yu et. al. [18] add another intermediate layer for multi-attribute based image retrieval which corresponds to a large pool (i.e. 6000) of weak attributes. Weak attributes are comprised of automatic classifier scores or other mid-level representations that can be easily acquired with little or no human labor. Chen et. al. focus on learning a regression model which introduces a cumulative attribute representation [19]. In details, each attribute is not only discriminative but also cumulative such that all other attribute values depend on their relative positions in a scalar value.

Human efforts involved in the class-attribute relationship designing are costly to obtain, subjective for evaluation and not scalable to large-scale datasets. Given images with category labels, Yu et.al. [20] formalize a category-attribute co-occurrence matrix for cross-category generalization. This is different from randomly generating ‘category splits’ in those geometric properties of category separability and attribute learnability are used.Chen et. al. [23] build facial classifiers which are based on appearance similarity of people with the same birth name. Another work of human description by visual attributes is proposed by Sadovnik et. al. [24]. The task is to describe a person in a group that distinguishes her from the others. The description will contain as minimum number of visual attributes as possible while it is maximizing the likelihood that a listener will correctly guess which person description refers.

3. UNSUPERVISED FEATURE LEARNING SCHEME

The overall flow chart of the visual recognition in this work can be basically split into three stages: Unsupervised data representation via attributes at mid-level, category based domain modeling, and evaluation of the classification performance. Given the dataset X = $\{x^{(i)} \mid i=1,2,3,\ldots,N\}$; where N is the number of train instances and $x^{(i)} \varepsilon R^d$ represents the low-level feature vector, we first divide it into three non-overlapping subsets randomly. Train set, $X_{train} = \{x^{(j)}, y^{(j)}\}_{j=1}^{K}$; where $X_{train} \subset X$, $y^{(j)} = \{1,2,3,\ldots,C\}$, and C is the number of classes in X_{train}, hold for the class label, is used in both unsupervised attribute learning and category modeling. Test set, $X_{test} \subset X$, is utilized in classifier evaluation while the free parameters of classifiers (i.e. kNN, SVM and Decision Tree) are optimized in a grid search method on a validation set, $X_{validation}$. Also note that $X_{validation}$ is achieved by inserting small amount of white noise to the samples in the dataset.

As mentioned earlier, we learn the classifier discriminants in a new feature space as the mid-level data representation, instead of simply using low level features. So we learn class based binary and relative attribute models in an unsupervised manner which will be detailed in subsections A and B, respectively. The binary attributes define output of the binary SVMs where the scores are computed by dot (i.e. scalar) products of the input samples, $x^{(j)}$, and the learned weight vectors, $w \varepsilon R^d$, of the binary SVMs. We call it Score Related Attribute (SRA) space. On the other hand, the relative attributes are modeled in Newton algorithm of [6]. Although this resembles of SVM method very much, the input signal would be the difference of related feature vectors, and the comparative condition determines the positive and negative sides, instead of tagging binary instance labels.

After we define binary and relative attributes in an unsupervised manner and learn them on the train set, X_{train}, we then model our classifiers; kNN, SVM and C4.5 decision tree still on the very limited set of train set. $X_{validation}$ is used to optimize the parameters of classifiers at hand whereas we compute the accuracy performance on the X_{test}, eventually. The flow chart of the proposed work is depicted in Fig. 2.

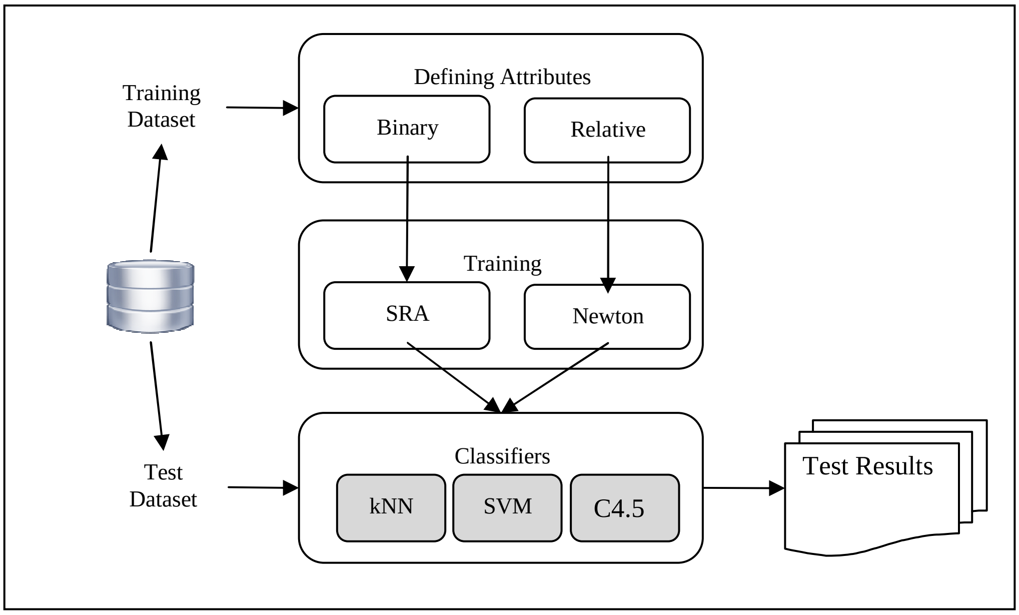

Figure 2: Schematic flow chart of the methodology

A. SRA Representation with Random Binary Attributes

We start with binary attributes for unsupervised feature learning. As the name explains itself, a binary attribute refers to whether it exists or not in the visual data. As speaking of class based attributes, we generalize them throughout each class specifically. For example in the statement “Attribute a_m exists in Class A, but Class B does not have it”, we hypothesize that all instances in a class contains or does not contain the mentioned attribute, a_m, at all. Although it seems to be more convenient if the attributes are assigned per instance individually, the literature work [6, 15] claim that class based attribute definitions result in consistency for learning the attribute models at the mid-level. Additionally, instance based labeling would consume much more effort while this process also hinders the unsupervised learning of data representation.

In detail, we define class based binary attributes randomly for the unsupervised learning in mid-level feature representation with some constraints to the random binary sequence generation. First, all ones or all zeros for an attribute $a_m = \{0,1\}^C$, is discarded from the list because they do not help us discriminating the visual classes in the attribute space. Next, we include random binary sequences into the consideration only if they have at least two positions different from each pattern which has been added to the list of attribute definitions already. Note that the number of positions which is set for discrimination is strongly related to the number of classes in the train set, X_{train}, since the length of each attribute sequence equals to it, C. Moreover, we explicitly limit the number of random sequences out of all combinations, 2^C. Nevertheless, binary definitions are produced as many as possible, and we select some of them for training, randomly. After we finish the random definitions, this process results in M class based binary attributes $A = \{a_m \mid m = 1,2,3,\ldots, M\}$, and we carry on the next step of modeling the attributes in binary SVMs.

SVM is a powerful tool for supervised learning which separates the feature space linearly into two categories: positive and negative. It tries to maximize the margin between positive and negative sides [35, 36]. The margin in SVM represents the gap between support vectors of both sides which are data samples acceptably close in limits to the opposite ones. The main idea is to find the optimum hyperplane that achieves the total minimum distance between the support vectors and the hyperplane. After training the SVM, a new sample is classified simply as either positive or negative by the result of the dot product with the optimized weight vector. In fact, SVM is a linear discriminant function and it obviously does not handle nonlinearly separable datasets with a satisfactory accuracy. Kernel functions (Gaussian, polynomial, chi-square, histogram intersection etc.) are introduced in the literature to establish nonlinear SVM classifiers and they achieve a justifiable popularity with SVM. Actually, SVM still preserves its linearity in this perspective but the input data are transferred into a new feature space via nonlinear kernel functions beforehand. Hence, we get more complex feature spaces with higher dimensionality instead of complicating the discriminant function itself.

In the second part of the introduced architecture, we propose binary attribute learning concept that is based on two-class SVM topology. Given the input data $X_{train} = \{x^{(j)},y^{(j)}\}$ and their class based binary attribute assignments $a_m = \{0,1\}^C$; input signals for the SVM are firstly achieved by Gaussian Kernel Function (GKF), $K(x^{(j)},X_{train})$:

$$K(x^{(j)},X_{train}) = e\frac{-\left\|x^{(j)} - X_{train}\right\|^2}{2\sigma^2} \; ; \; x^{(j)} \in X_{train}$$

(1)

where σ is the scale parameter that factors the neighborhood. So each data sample is now represented by its GKF responses to the data samples (i.e. landmarks) in X_{train}. Note that the input dimensionality now equals to the number of instances, K. Since SVM is a supervised learning algorithm, the sample-attribute assignments obtained in the previous class based attribute definition, $a_m : y^{(j)} \rightarrow \{0,1\}$, are now used as the data labels for supervision, instead of $y^{(j)}$ itself. The unconstrained objective function of the SVM is:

$$y^{(j)}(w^T x^{(j)} + b) \; > \; 1 \; ; \forall j$$

(2)

$$J_{G_X,y}(W,b) = P\frac{1}{K}\sum_{j=1}^{K} max(1\text{-}w^T G_{x^{(j)}} y^{(j)},0)^2 + \frac{1}{2}\sum\left\|W\right\|^2 ; y^{(j)} \in \{-1,1\}$$

(3)

where P is the trade-off constant, penalizing data points which violate the margin requirements. G_x represents GKF output vector of (1) for each sample, $x^{(j)}$, that is the new input signal to the SVM. W is the matrix, which embeds the parameter vectors, including biases, b. They are assumed to be orthogonal to the hyperplanes that separate both sides (i.e. binary assignments) and initialized randomly. As aforementioned, SVM simply separates the space into two parts. So the desired output signal for each input, $y^{(j)}$, is achieved by assigning 1 for the classes which have the attribute, a_m, and -1 for the rest. The stochastic gradient descent algorithm is then employed as:

$$h_W(G_x) = W^T G_x \tag{4}$$

$$\delta w = \frac{\delta J(W,b)}{\delta h_W} = \text{-}2Py\ max(1\text{-}h_W(G_x)y,0) \tag{5}$$

$$\Delta w = \alpha \left(\frac{1}{N} \sum (G_x^T \delta w) + w \right) \tag{6}$$

$$w_{new} = w_{old} - \Delta w \tag{7}$$

where δw is the back propagated derivation of the error signal per data sample, Δw is the average weight correction that includes L_2 regularization without bias terms. Also note that α is the learning rate and h_w is the hypothesis function of the SVM. Once the SVM is set up, we optimize the weight parameters iteratively. The hyperplane is updated with the max-margin objective function to separate the samples of each side based on the static sample-attribute assignments. The iteration is terminated when the saddle point is reached.

After we find optimum parameters of the SVM, w^T, the data sample, $x^{(j)}$, can simply be conveyed to the new feature space by its related binary attribute score with Score Related Attribute (SRA):

$$SRA(x^{(j)};w_{a_m}) = w_{a_m}^T x^{(j)} \tag{8}$$

where w_{a_m} is the weight vector of SVM which corresponds to the binary attribute definition a_m, including the bias term, b. So we train an independent SVM for each random binary attribute with the given train set, X_{train}, and the visual data is now in a new M (i.e. number of binary attributes) dimensional feature space by their SVM scores. Eventually, we implement our classifier algorithms on the train set X_{train}, where the feature vectors are now represented in the mid-level, instead of their original space R^d, where M << d.

B. Rank Based Representation with Relative Attributes

The relative attribute definitions are first introduced in [6] and they have attracted much attention so far [15, 16, 30]. Unlike binary attributes, they infer the relative strength of an attribute on the visual data, instead of exposing the existence (or non-existence). As it can be seen in the statement of "Class A has attribute a_m more than Class B, but less than Class C," the class based relative attributes order the visual categories on the basis vectors of a new feature space by comparative constraints; i.e. more/less than. They have obvious advantages over the binary definitions in those: 1. More input data are fed into the attribute learning models because the input data are now the pairwise comparisons of the samples. Assuming that each class has K examples and we have C categories in the training data set, X_{train}, then the number of input data will be $C\binom{C}{2}K^2$, instead of KC. So we assume that more training data would increase the accuracy performance in learning the attribute models. 2. Since we randomly define the relative attributes by ordering the classes in each attribute basis, the total number of possibly generated ordering patterns equals to the permutation of the number of classes, C. Thus, one can produce many random ordering sequences more than the binary predicates, and more discriminative patterns may be selected among them.

Given a class based ordering, $a_m = \{c^{(1)} > c^{(2)} > c^{(3)} > \ldots > c^{(i)}\}$; $c^{(i)} \in C$, which relates every category to each other with a less/more condition, we use the Newton method of [6] for a relative attribute as:

$$r_m(x^{(j)}) = w_m^T x^{(j)}$$

(9)

$$\forall (i,j) \in O_m : w_m^T x_{C_a}^{(i)} > w_m^T x_{C_b}^{(j)} \ ; \ i \in c_a \ , j \in c_b \ , \ c_a > c_b$$

(10)

$$w_m^T(x^{(i)} - x^{(j)}) \geq 1\text{-}\gamma_{ij} \quad ; \ \forall (i,j) \in O_m \ , \gamma_{ij} \geq 0$$

(11)

$$argmin_{w_m}(\frac{1}{2} \left\| w_m^T \right\|_{L_2}^2 + T \sum \gamma_{ij}^2$$

(12)

where r_m is the reel ranking score of the training instance, $x^{(j)}$, on the attribute basis, a_m, $w_m \in R^d$ is the parameter vector of the relative attribute model, O_m is the set which consists of pairwise data instances holding for the more/less conditions. When we look into (11) closer, the equation is very similar to that of the SVM. But the input signal is now the difference of pairwise feature vectors from the set, O_m, not the low-level feature vectors itself. So the optimum solution would then order the classes on the weight vector, w_m, by minimizing the cost function of (12); where T is the constant that regulates the balance between weight decreasing and the non-negative slack variables, γ_{ij}. This results in maximizing the margin between classes in the order definition, a_m.

Once we optimize the free parameters, w_m, the attribute strength is computed as in the binary attribute score. Hence, we convey the original input data $x^{(j)}$ into a mid-level feature space by M (i.e. number of generated relative attributes) dimensional ranking scores, M << d. The next step is to answer how one may generate class orderings for relative attribute modeling which will be detailed in the subsections below.

(1) Random Relative Attributes

We follow the same approach of binary attribute generation described in section 3.A. The relative definitions, $A = \{a_m \mid m = 1,2,3,\ldots, M\}$, indicate ordering the visual categories, $\{c^{(1)} > c^{(2)} > c^{(3)} > \ldots > c^{(i)}\}$; $c^{(i)} \in C$, randomly for each attribute, a_m. The class ordering expands the feature space much more than binary attributes and we have many options this time. The random class based ordering sequences are included into the consideration only if they have at least four positions different from each pattern which has been added to the list of attribute definitions already. Note that the number of different positions is twice that of binary predicates. So we produce many unique orderings (say 1,000) and select M sequences out of them randomly.

(2) Selective Relative Attributes

To make attribute definitions more discriminative, we propose a new approach for picking some orderings based on Kendall Tau (KT) correlation metric [15], instead of selecting randomly. For each pair of randomly generated attribute definition, KT is computed as:

$$KT = \frac{n_c - n_d}{C(^{C}_{2})}$$

(13)

where n_c and n_d are the number of concordant and discordant pairs between the two orderings and the denominator refers to the total number of pairs. The range of KT is then in [-1,1], and it is -1 if two orderings are complately different (1 if they are the same). Thus, we first compute a correlation matrix in which each element is the KT value of pairwise orderings of all generated ones, next the average correlation values of all definitions are sorted in the decreasing order, finally we select the top M random orderings (i.e. least correlated) among them. Thereafter, the preselected classifiers (i.e. kNN, SVM and C4.5 decision tree) are modeled on the train set, X_{train}, while optimizing their free parameters with $X_{validation}$.

4. EVALUATION OF THE PROPOSED WORK

A. Experimental Setup

We use Outdoor Scene Recognition (OSR) Dataset [6] containing 2,688 images of 8 scene categories. The distribution of images for the dataset is shown at Table I. Note that the number of samples in each class varies. OSR dataset is also utilized in [6] and [15] which are the recent studies in attribute based object recognition literature. Besides, the provided low level features (i.e. GIST) and the same train/test splits for multiple runs are used as the initial input in multi-category classification schemes. Most of the outdoor scenes in OSR dataset display large intra-class variability, meaning that object contents within a scene category are very different while inter-class variance is small especially for the natural scene categories. This issue makes the object classification problem harder when working with OSR dataset. Example images from OSR are displayed in Fig. 3, respectively.

Table I. THE DISTRIBUTION OF IMAGES FOR OSR DATASET.

OSR Dataset	Coast	Forest	Hihgway	Insidecity	Mountain	Open Country	Street	Tall Building
	360	328	260	308	374	410	292	356

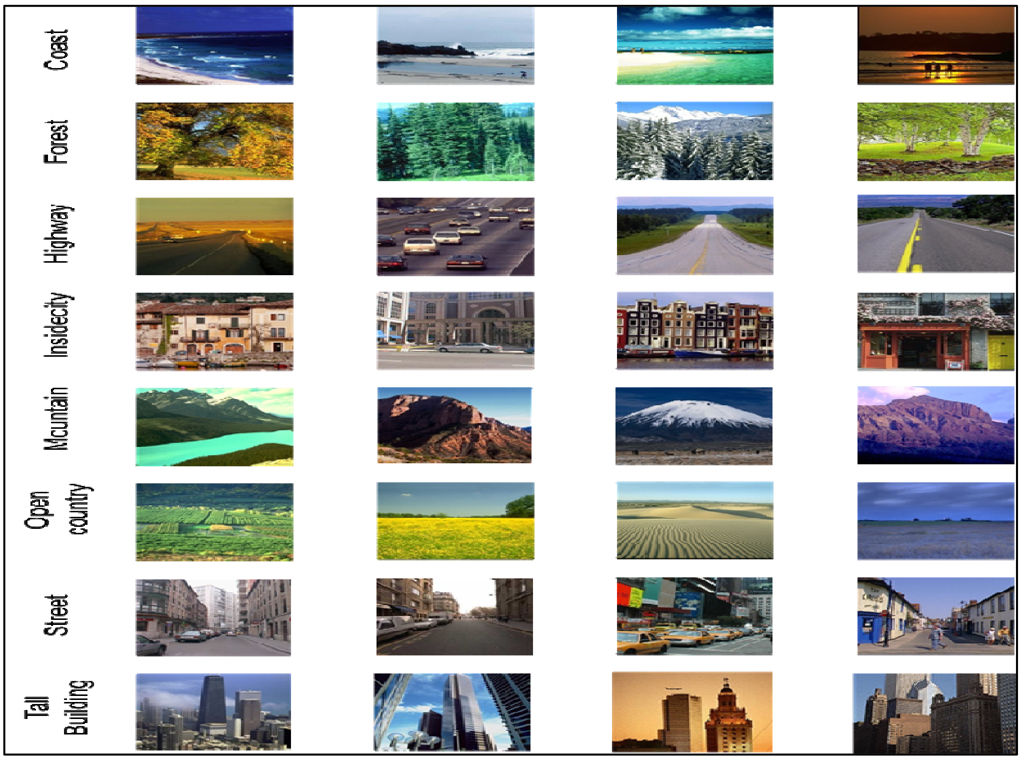

Figure 3: OSR dataset sample images.

For the training phase of both attribute models and visual classes, we randomly select 30 instances from each class as X_{train}, and the rest is used as the test set, X_{test}. Note that the X_{train} is very limited due to the mid-level attribute representation when compared to the low-level features in classification. The experiments are repeated 20 times, and the mean and standard deviation values are noted at tables for comparative results whereas the average accuracies are used in the figures. Additionally, we limit the number of both randomly generated relative and binary attributes to 28 for the sake of comparison to the other literature work.

Furthermore, we evaluate three algorithms to measure their classification accuracies in the mid-level attribute space: SVM, kNN, and C4.5 decision tree. We select these methods as they are powerful and popular discriminants on the shelf. So WEKA toolbox [37] is used to implement them while we optimize their free parameters (i.e. the regulator constant, C, for SVM; the number of nearest neighbors, k, for kNN; the pruning confidence, C, and the minimum number of samples, M, for the decision tree) on the $X_{validation}$. Additionally, we normalize the feature vectors of attribute scores as the new inputs to the classifiers by whitening process of [38] in order to achieve zero mean and unit standard deviation for each dimension.

Finally, we also use the supervised binary and relative attribute definitions which are given in [6] to promote the benefits of unsupervised (i.e. randomly generated) definitions. Fig. 4 displays the usage combinations of all attribute patterns that are utilized for the experiments, detailed in the next subsection.

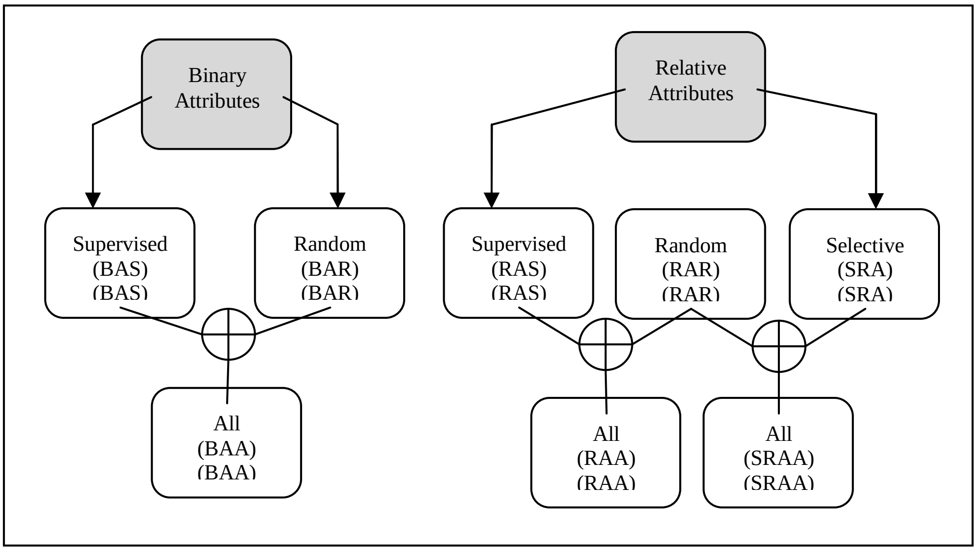

Figure 4: Attribute based comparisons scheme.

B. Classification Results

In this subsection, we first analyze the classification results in different configurations of attributes and classifiers. The experiments are repeated 20 times and the mean and standard deviations are noted for binary attributes as SRA results and relative attributes as Newton ranking scores at Table II and III, respectively. Also note that we use the 6 binary and relative attribute definitions which are already established on OSR dataset in a supervised manner [6]. Additionally, we generate/select 28 random binary and relative attributes to compare the classification results with the other literature work, although we may produce them as many as needed that is dependent with the number of visual classes at hand.

Table II. SRA RESULTS WITH BINARY ATTRIBUTES ON OSR DATASET.

	Classifier Accuracies (%)		
Attribute Type	**kNN**	**Decision Tree**	**SVM**
binary_attributes_supervised (BAS)	52.99 ± 2.58	49.07 ± 2.32	54.88 ± 3.68
binary_attributes_random (BAR)	75.27 ± 1.52	62.31 ± 3.99	74.28 ± 1.63
binary_attributes_all (BAA)	76.38 ± 1.55	64.38 ± 3.34	**76.73 ± 1.59**

For speaking of supervision, randomly generated attributes outperform the human labeled attributes at both tables considerably. This is due to the fact that we can generate more definitions randomly at no cost and this expands the mid-level feature space discriminatively which results in better accuracies. We claim that supervision may sometimes divert the learning system into a worse situation as it is subject to the human experience, and hard work of labeling. Nevertheless, we can surely append the supervised attributes into the unsupervised patterns if they exist. We achieve almost 2 % increase in the performance at both tables when they are concatenated with the unsupervised attributes. Additionally, relative attributes overcome the binary definitions about 2-3 %. We assume that the class orderings which we may produce randomly is related to the permutation of the number of categories, not power of 2, and that gives many more choices for selection. Additionally for the relative attributes as detailed in section 3.b, we run the KT algorithm to select more discriminative ordering patterns from the randomly generated pool, instead of random selection. We see that selective relative attributes indeed increase the performance more than 2 %, and this confirms our previous assumptions. On the other hand, SVM algorithm achieves better accuracies than kNN and C4.5 decision tree overall while C4.5 is the worst. Note that the kNN gets the similar, even better results than SVM although it is the simplest instance-based classifier. We assume that non-parametric learning of the kNN method benefits the attribute based feature space more than the others.

Table III. NEWTON RESULTS WITH RELATIVE ATTRIBUTES ON OSR DATASET.

	Classifier Accuracies (%)		
Attribute Type	**kNN**	**Decision Tree**	**SVM**
relative_attributes_supervised (RAS)	62.17 ± 1.02	54.54 ± 2.05	63.12 ± 1.79
relative_attributes_random (RAR)	76.78 ± 1.97	69.77 ± 2.63	76.34 ± 1.52
relative_attributes_all (RAA)	77.15 ± 1.52	73.57 ± 2.04	77.86 ± 2.13
selective_relative_attributes (SRA)	77.24 ± 1.87	70.81 ± 3.62	77.12 ± 1.96
selective_relative_attributes_all (SRAA)	78.36 ± 2.01	72.66 ± 2.91	**79.86 ± 2.52**

Next, the proposed method is compared with the similar approaches in literature on the same experimental setups, and the mean accuracy results of the multiple experiments are listed at Table IV. BINs, PCA and FLD algorithms are actually used for dimension reduction and these references are not related to the attribute learning. Nevertheless, the basis vectors (i.e. like attribute weight vectors, w) which are extracted during the implementations help representing the data in a new features space, so they are included as baselines for this reason. Besides, the other methods generate supervised/unsupervised attributes in the intermediate level for visual recognition, like the proposed work.

The results at in Table IV show that the proposed method outperforms the other approaches for about a minimum of 1 % with the selective relative attributes. In general, it is observed that the attribute-based methods achieve much better accuracies than the other baseline works. So the attributes do not only reduce the dimensionality but also do they constitute a more representative space in the mid-level. On the other side, the unsupervised attributes display increased performance when compared with the supervised ones. Additionally, the accuracy rises up even further when we combine the both types. We assume that the expanded number of unsupervised attributes with distinct class orderings establish a better representation without human laboring, leading to more effective classifiers.

Table IV. PERFORMANCE COMPARISON OF THE ALGORITHMS.

Algorithms	**# of Attributes**	**Mean Accuracy (%)**
BINs [15]	28	76.05
PCA [15]	34	71.46
FLD [15]	28	63.10
Supervised Attributes (SAT) [6]	6	72.82
Unsupervised Attributes (UAT) [15]	28	76.57
SAT+UAT [15]	34	77.88
RAS [16]	34	78.64
Our Binary ALL	34	76.73
Our Selective Relative ALL	34	**79.86**

Additionally, we evaluate the behavior of mid-level feature space by changing the number of attributes that we generate randomly in the proposed work, and the graphical results are displayed in Fig. 5. Note that we use the SVM results as it is better than kNN and C4.5, comparatively. When we take into account the results of supervised attributes (i.e. 6 binary/relative definitions) at Table II and III, the accuracy performance is almost the same with 10-12 randomly generated binary and relative attributes, respectively. After this point, we outperform the supervised attributes obviously, and it confirms that the performance is increased as we enlarge the feature space with more attributes, although we select them randomly. Moreover, the relative definitions achieves better accuracies than the binary attributes. We think the main reason is that the Newton method orders the visual categories by maximizing the sequential margins with many more pairwise inputs, and we can generate more orderings than the binary predicates. Also, the selective relative attributes gets the best performance since the KT correlation metric is used to pick the more distinctive orderings, instead of simply selecting them randomly. Another point is that we can have even better results if the supervised attributes are concatenated with the unsupervised orderings. One may use the unsupervised definitions as the supplementary feature space if the supervised attributes already exist.

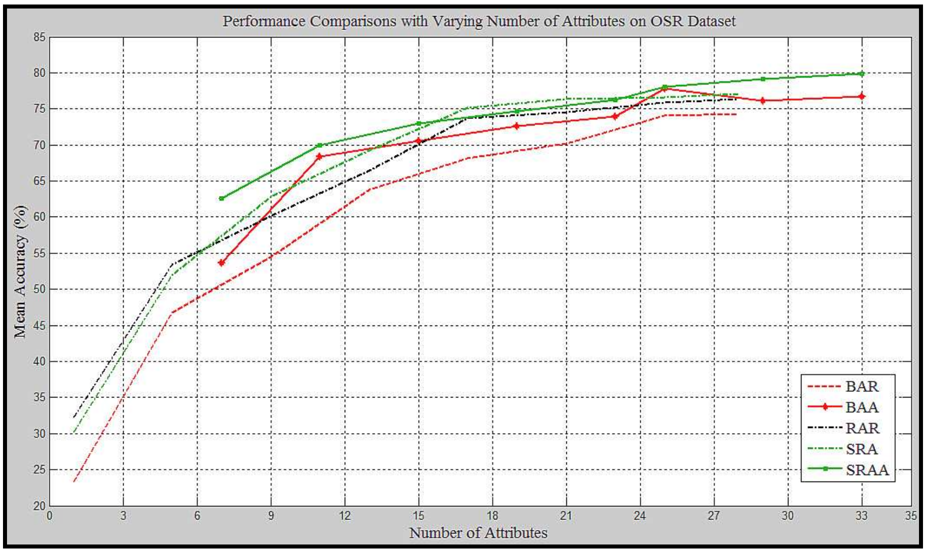

Figure 5: Performance comparisons with varying number of attributes.

Receiver Operating Characteristic (ROC) curve is frequently used in literature to evaluate the performance of classifiers. Basically, the ratio of false and true positive samples is plotted by changing thresholds in a step-wise manner. The classifier is regarded as more successful when its plot rises up earlier and sharper than the others. Eventually, we compare the performances of binary, relative and the selective attributes with their supervised and combined (i.e. supervised + unsupervised) versions on ROC curves for OSR dataset in Fig. 6. As seen, the accuracy is increased obviously when all attributes are used together, and this confirms that the unsupervised attributes add discriminative power in dimensionality. Additionally, the selective relative attributes outperform the others clearly while relative definitions are better than the binary predicates.

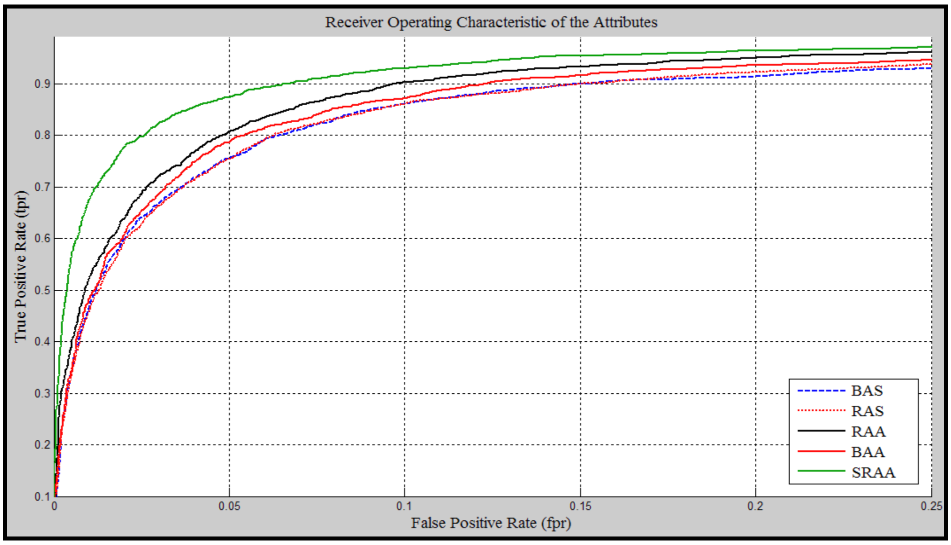

Figure 6: ROC analysis of the attribute types on OSR dataset.

5. CONCLUSION

In this work, we introduce two approaches for the mid-level visual data representations in an unsupervised manner which is based on the binary and relative attributes, respectively. Binary attributes mainly split the low-level feature space into two sides; i.e. positive and negative. Then, the SVM algorithm is established to maximize the margin, and its scores are used as the new data representation. On the other hand, the Newton method tries to maximize the gap between visual categories based on a definition which describes the relative ordering. So we first generate random attribute definitions with some limited constraints that assure to get exclusively different binary predicates and relative orderings. Thereafter, we convey the low-level feature vectors into a more discriminative attribute space by using their new representations, and the classification is carried on this new space.

In the experiments, we utilize a mid-scale visual recognition dataset, OSR, to evaluate the combinational attribute types and classifiers, namely SVM, kNN, and C4.5 decision tree. Also note that only a limited set of train data is used for learning both the attribute and classification models which benefits the mid-level data representation. The results reveal that the unsupervised attributes outperform the supervised definitions clearly although we produce them randomly without any effort. Additionally, KT correlation metric is used to pick the more discriminative orderings among randomly generated sequences, instead of simply selecting them randomly. This also boosts the accuracy performance slightly. Moreover, we have even better results if the supervised attributes are concatenated with the unsupervised orderings. We conclude that the unsupervised definitions can be used as the supplementary features if the supervised attributes already exist.

For the future work, we focus on the relative attribute selection issue since it already proves to be an important tool for the performance increase. Also, the classifier algorithms can surely be used in a combined form, called mixture of experts, to make better decisions at the end of the classification process. Lastly, an incremental learning scheme can be established on the proposed work which refers to learning the attribute space and category models simultaneously in an iterative way.

REFERENCES

[1] Ferrari V. and Zisserman A. “Learning visual attributes” Advances in Neural Information Processing Systems, Vancouver CA, December 2007.

[2] Lampert C.H., Nickisch H. and Harmeling S. “Attribute-Based classification for zero-shot visual object categorization” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 36, no. 3, 2014.

[3] Lampert C. H., Nickisch H., and Harmeling S. "Learning To Detect Unseen Object Classes by Between-Class Attribute Transfer" Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2009.

[4] Farhadi A., Endres I. and Hoiem D. “Attribute-centric recognition for cross-category generalization” CVPR, 2010.

[5] Farhadi A., Endres I., Hoiem D. and Forsyth D. “Describing objects by their attributes” CVPR 2009.

[6] Parikh D. and Grauman K. “Relative attributes” Int’l Conference on Computer Vision (ICCV), 2011.

[7] Sharma G., Jurie F., and Schmid C. “Expanded parts model for human attribute and action recognition in still images” CVPR, pp. 652 – 659, 2013.

[8] Akata Z., Perronnin F., Harchaoui Z. and Schmid C. “Label-embedding for attribute-based classification” CVPR, pp. 819 – 826, 2013.

[9] Tamara L.B., Alexander C.B. and Jonathan S. “Automatic attribute discovery and characterization from noisy web data” ECCV, pp. 663-676, 2010.

[10] Russakovsky O. and Fei-Fei L. “Attribute learning in large-scale datasets” ECCV Workshops, pp. 1-14, 2010.

[11] Biswas A. and Parikh D. “Simultaneous active learning of classifiers & attributes via relative feedback” CVPR, 2013.

[12] Parkash A. and Parikh D. “Attributes for Classifier Feedback” European Conference on Computer Vision (ECCV), vol. 3, pp. 354-368, 2012.

[13] Rastegari M., Diba A., Parikh D., Farhadi A. “Multi-attribute queries: To merge or not to Merge” CVPR, 2013.

[14] Kumar N., Berg A.C., Belhumeur P. N., and Nayar S. K. “Attribute and smile classifiers for face verification” ICCV, 2009.

[15] Ma S., Sclaroff S. and Cinbis N.I. "Unsupervised learning of discriminative relative visual attributes" ECCV Workshop on Parts and Attributes, 2012.

[16] Karayel M. and Arica N. “Random attributes for image classification” IEEE 21th Conference on Signal Processing and Communications Applications, 2013.

[17] Wang Y. and Mori G. “A discriminative latent model of object classes and attributes” ECCV, pp. 155-168, 2010.

[18] Yu F.X., Ji R., Tsai M., Ye G. and Chang S. “Weak attributes for large-scale image retrieval” CVPR, 2012.

[19] Chen K., Gong S., Xiang T. and Loy C.C. “Cumulative attribute space for age and crowd density estimation” CVPR, pp. 2467 – 2474, 2013.

[20] Yu F.X., Cao L., Feris R.S., Smith J.R. and Chang S. “Designing category-level attributes for discriminative visual recognition” CVPR, 2013.

[21] Li W., Yu Q., Sawhney H. and Vasconcelos N. “Recognizing activities via bag of words for attribute dynamics” CVPR, pp. 2587 – 2594, 2013.

[22] Ma Z., Yang Y., Xu Z., Sebe N., Yan S. and Hauptmann A.G. “Complex event detection via multi-source video attributes” CVPR, 2013.

[23] Chen H., Gallagher A. and Girod B. “What's in a name: first names as facial attributes” CVPR, 2013.

[24] Sadovnik A., Gallagher A. and Chen T. "It's not polite to point: describing people with uncertain attributes" CVPR, 2013.

[25] Choi J., Rastegari M., Farhadi A. and Davis L.S. “Adding unlabeled samples to categories by learned attributes” CVPR, 2013.

[26] Wah C. and Belongie S. “Attribute-based detection of unfamiliar classes with humans in the loop” CVPR, pp. 779 – 786, 2013.

[27] Wang S., Joo J., Wang Y., and Zhu S.C. “Weakly supervised learning for attribute localization in outdoor scenes” CVPR, 2013.

[28] Saleh B., Farhadi A. and Elgammal A. “Object-centric anomaly detection by attribute-based reasoning,” CVPR, 2013.

[29] Bosch A., Xavier M. and Marti R. “A review: which is the best way to organize/classify images by content?” Image and Vision Computing, 2006.

[30] Ergül E., Ertürk S. and Arica N. “Unsupervised Relative Attribute Extraction” IEEE 21th Conference on Signal Processing and Communications Applications, 2013.

[31] Chang C.C. and Lin C.J. “LIBSVM : A library for support vector machines” ACM Transactions on Intelligent Systems and Technology, pp. 1-27, 2011.

[32] Shrivastava A., Singh S. and Gupta A. "Constrained semi-supervised learning using attributes and comparative attributes", ECCV, vol 3, pp. 369-383. 2012.

[33] Yu, A., and Grauman, ,K., “Just Noticeable Differences in Visual Attributes” ICCV, 2015.

[34] Verma, Y., and Jawahar, C.V., “Exploring Locally Rigid Discriminative Patches for Learning Relative Attributes” ICCV, 2015.

[35] Alpaydın E., “Support Vector Machines,” in Introduction to machine Learning, The MIT Press, London, 2004, pp. 218-225.

[36] Cortes C. and Vapnik V., "Support-vector networks," Machine Learning, vol. 20, no. 3, pp. 273-297, 1995.

[37] Waikato University, Waikato Environment for Knowledge Analysis (Weka) Versiyon 3.7.11, Waikato University, Hamilton, 2014.

[38] Coates A., Lee H. and Andrew Y. Ng. “An analysis of single-layer networks in unsupervised feature Learning,” International Conference on Artificial Intelligence and Statistics (AISTATS), 2011.