T.R.
TURKISH NAVAL ACADEMY
NAVAL SCIENCE AND ENGINEERING INSTITUTE
DEPARTMENT OF COMPUTER ENGINEERING

SYNTHETIC ATTRIBUTES FOR IMAGE CLASSIFICATION

MASTER THESIS

MEHMET KARAYEL

Advisor: Assoc.Prof.Dr. Nafiz ARICA

İstanbul, 2013

Synthetic Attributes For Image Classification

MEHMET KARAYEL

Submitted partial fulfillment of the requirements for degree of

MASTER OF SCIENCE IN COMPUTER ENGINEERING

Turkish Naval Academy
Naval Science and Engineering Institute

Author: ----------
Mehmet KARAYEL

Defence Date:

Approved by: ----------
Assoc.Prof.Nafiz ARICA (Thesis Advisor)

Prof.Dr. Cem ÜNSALAN (Defense Committee Member)

Assoc.Prof. Emin Erkan KORKMAZ (Defense Committee Member)

DISCLAIMER STATEMENT

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Turkish Naval Forces, Turkish Naval Academy and Naval Science and Engineering Institute.

DEDICATION

To my family, KARAYEL.

ACKNOWLEDGEMENT

I would like to render my spatial thanks to

My family, Azime, Nermin, Selin KARAYEL, for their moral everlasting support and giving me enthusiasm,

My advisor, Assoc. Prof. Nafiz Arıca, for giving me his ultimate guidance and spirit,

My friend, Haydar YEŞİL, for his helps and unique patience,

Institute director, Mustafa Karadeniz, for his active collaboration,

My anonymous tutors for their priceless knowledge.

This thesis would not have been possible without their trust on me.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS AND ABBREVIATIONS

AMT	:	Amazon Mechanical Turk
DAP	:	Direct Attribute Prediction
FLD	:	Fisher's Linear Discriminant
IAP	:	Indirect attribute Prediction
MAP	:	Maximum a Posteriori
mRMR	:	Maximum Relavance Minimum Redundancy
MI	:	Mutual Information
NormMI	:	Normalized Mutual Information
KNN	:	K-Nearest Neighbours
PAC	:	Pricipal Component Analysis
RBF	:	Radial Basis Function
RGB	:	Red, Green, Blue
SVM	:	Support Vector Machine

ÖZET

İMGE SINIFLANDIRMA İÇİN SENTETİK NİTELİKLER

Mehmet KARAYEL

Bilgisayar Mühendisliği M.S. Tezi, 2013

Danışman: Doç. Dr. Nafiz ARICA

Anahtar Kelimeler: *Sentetik Nitelikler, Rastgele Nitelikler, Nitelik Seçimi, Nitelik Tabanlı Yaklaşımlar, İmge Sınıflandırma.*

Bilgisayarla görü alanındaki en önemli problemlerden birisi olan imge sınıflandırma için öznitelik tabanlı klasik yaklaşımların yanı sıra nitelik tabanlı yaklaşımlar son yıllarda sıklıkla kullanılmaya başlamıştır. Nitelik tabanlı yaklaşımların en önemli avantajı, insanlar için anlam ifade eden niteliklerin kullanılması vasıtasıyla insanoğluna benzer bir öğrenme yapılabilmesidir. Ayrıca nitelik tabanlı yaklaşım sayesinde, eğitim aşamasında görülmeyen imgeler sınıflandırılabilmekte veya sınıflandırma yapılamasa dahi imge özellikleri hakkında bilgi sahibi olunabilmektedir. Ancak, imgeler ile ilişkili niteliklerin gözetimli veya gözetimsiz olarak belirlenmesi aşamasında halen birçok problem ile karşılaşılmaktadır.

Bu tezde mevcut nitelik tabanlı imge sınıflandırma problemine yönelik olarak “Sentetik Nitelikler” yaklaşımı önerilmektedir. Sentetik nitelikler, imgeleri

betimleyen farazi nitelikler olarak tanımlanabilir ve öznitelik uzayından otomatik olarak çıkarılır.

Bu amaçla tezin ilk bölümünde önceden belirlenen nitelik sayısına bağlı olarak oluşturulan nitelik uzayındaki olası tüm nitelikler arasından rastgele nitelikler çıkarılmaktadır. Rastgele nitelikler daha sonra tek tek öznitelik uzayında eğitilir. Dolayısıyla nitelik belirleme işlemi literatürdeki gözetimli ve gözetimsiz yaklaşımlardan farklı olarak zahmetsiz bir şekilde gerçekleştirilmektedir.

Tezin ikinci bölümünde ise rastgele niteliklerin daha mantıklı bir şekilde çıkarılması üzerinde yoğunlaşılmaktadır. Bu amaçla imge sınıfları ile nitelikler arasındaki bağlantı kullanılarak, imgeleri betimleyen ayırt edici nitelikler seçilmektedir. Sentetik nitelik olarak adlandırdığımız bu yaklaşımda, ikili nitelik seçimi sadece bir imge sınıfına özel niteliklerden başlanarak sırasıyla daha çok imge sınıfının betimlenmesinde kullanılabilecek niteliklerin eklenmesi suretiyle gerçekleştirilmektedir. Göreceli sentetik nitelikler ise rastgele nitelikler arasından sıra birleştirme yöntemi ile belirlenmektedir.

Önerilen her iki yaklaşım literatürdeki diğer nitelik tabanlı çalışmalarla aynı veri kümesi üzerinden test edilerek karşılaştırılmaktadır. Yapılan deneylerde diğer çalışmalarda elde edilen en yüksek imge sınıflandırma performanslarına ulaşılmaktadır.

ABSTRACT

SYNTHETIC ATTRIBUTES FOR IMAGE CLASSIFICATION

Mehmet KARAYEL

Computer Engineering M.S. Thesis, 2013

Advisor: Assoc.Prof.Dr. Nafiz ARICA

Keywords: *Synthetic Attributes*, *Random Attributes, Attribute Selection, Attribute-Based Approaches, Image Classification.*

Attribute based approaches have been employed frequently in recent years in addition to classical approaches based on low level features for image classification which is one of the most important problems computer vision. The most important advantage of this approach is that learning can be performed similar to human by using attributes which makes sense for people. In addition, images which are not seen in training phase can be classified or even if correct classifying is not applicable, it is possible to obtain information about the image properties owing to attribute based approach. However, there are many problems in generation of supervised or unsupervised attributes which are associated with image classes.

In this thesis, we propose “Synthetic Attributes” approach to solve the problems of current attribute based image classification. Synthetic attributes

can be described as hypothetical attributes which are extracted from feature space automatically.

For this purpose, in the first part of thesis, random attributes are extracted from among all possible attributes which are created depending on predetermined number of attributes in attribute space. Random attributes are then trained in feature space particularly. Therefore, the process of generating attributes is realized easily unlike supervised and unsupervised approaches in the literature.

In the second part of thesis, it is focused on extracting random attributes more consciously. For this purpose, discriminative attributes depicting images are selected by using the relation between image classes and attributes. In this approach called synthetic attributes, binary attribute selection is carried out by starting with only the attributes which are specific for one class and adding the attributes which can be used for depicting other image classes. Relative synthetic attributes are also determined among random attributes by rank aggregation methods.

The proposed approaches have been tested and compared to the other attribute based studies in the literature on the same data sets. The highest image classification performances obtained in other studies has been reached in the experiments.

CHAPTER 1

INTRODUCTION

Computer vision is a field that includes methods for acquiring, processing, analyzing, and understanding visual data. It is fundamental for many intelligent systems, because one of the most important features that make a system intelligent is to obtain meaningful and useful conclusions from image or video like human brain system. The classification and identification of objects and discovering the relations between them emerges as a preliminary requirements for gathering meaningful conclusions which can be used by intelligent systems.

However, image/object classification is an open research area in computer vision. In order to classify the image/object categories efficently, various methods have been proposed for over 40 years in this field. These methods can be grouped as; 1) Geometric based approaches that focus on obtaining the geometric model of objects in different enviromental conditions, 2) Global modelling approaches that use descriptors based on pixel values, 3) Object part based approaches that aim to achieve the whole object by detecting interest points, 4) Region based approaches that create models on the basis of segmentation results.

If we analyze the development of image/object classification, it can be observed that the proposed approaches have been transformed into more abstract and semantic architecture in time to approximate human learning. In recent years, attribute based approach which is much similar to the nature of

human learning is being used extensively for image/object classification. Human beings can easily describe and give a meaning to all different object categories by their heritable superior capacity of abstraction and categorization. Thus, human can perform abstraction, classification and categorization by using very few examples. Due to the nature of objects, different object categories can share the same characteristics. For example, “striped” attributes can be learned by using instances of zebra, tiger, bee and shirts with stripes. Owing to these important property of attributes, fewer examples are needed during training phase in attribute based classification.

Furthermore, the representation of unknown images which are not seen in training phase can be realized by the attribute based approaches. Even if correct classifying is not possible, the information about the properties of objects can be obtained and learning can be performed using textual descriptions.

In attribute based approach, image/objects are described with “attributes”, which is defined as “an inherent characteristic” of an object in Webster’s dictionary. Attributes are more meaningful to human compared to low level features which do not make any sense. They can be divided into two classes as visual and semantic. Visual attributes include color, shape and texture properties of images. Visual attributes have a general structure and their discriminative capacity is low compared to semantic atrributes in object classification. Semantic attributes include functional, structural and hereditary characteristics and any kind of semantic properties of images. Semantic attributes are more specific, abstract and discriminative compared

to visual attributes. Therefore, two instances from different object categories having similar visual attributes can be distinguished by semantic attributes.

Image attributes can be represented as binary or relatively. While binary attribute representation reflects the notion of “presence or absence of attribute in an image”, relative attribute represents “the strength of attribute in an image”. In binary description, classical classifiers which find the maximum margin hyperplane for discriminating the image instances in feature space are used for obtaining the presence or absence of attribute in an image. In relative description, an optimization function is employed for sorting images with respect to the correponding attribute. Learning stage of relative attributes are performed by maximizing the margin between the image instances after projecting them into a line.

Attributes can be learned by using supervised or unsupervised way. In supervised attribute learning, images are labeled with attributes by human effort and many difficulties occur due to human factor. These diffuculties can be summarized as; 1) Usually more general and intiutive attributes are determined instead of discriminative attributes which are appropriate for classfication purposes, 2) Some discriminative attributes may be overlooked or could not be expressed by words, 3) Erroneous attribute tagging can be needed, 4) The process of attribute extraction become exhaustive and it takes a long time in large datasets which contain many attributes.

In addition to above difficulties; attribute labeling of datasets in supervised methods needs a great deal of effort and budget. If we extract attributes by searching images on the Internet search engines, the resulting

attributes can be irrevelant with the image categories. As a result, supervised attribute determination is very exhausting and needs too much effort.

Unsupervised attribute learning, on the other hand, skip the attribute generation stage and extract attributes directly in feature space of images. To our knowledge, there are only two unsupervised studies for binary attribute learning [Kumar. 2009] [Farhadi, 2009]. In both studies, unsupervised binary attributes are used in addition to the supervised binary attributes. In [Kumar. 2009], hand-labeled dataset is used for creating the unsupervised binary attributes in order to improve the classification accuracy. In [Farhadi, 2009], unsupervised binary attributes are created only for the object classes which are not distinguished by visual and semantic attributes.

In unsupervised relative attribute learning, we found only one study [Ma, 2012]. In this study, attributes are learned after some internal operations in feature space and attribute learning is performed on the basis of entropy criteria related to the number object classes. In order to determine the attributes, too many operations are performed in feature space. During the process of learning attributes, search space can be enlarged exponentially with respect to the number of classes. Even if the attribute extraction operation is made over the image instances instead of image classes, the enlarging problem in the search space will still be unavoidable. In addition, it is evaluated that the criteria (entropy, etc.) which is used for ranking classes has no semantic meaning. Since the number of attribute is relative to the number of object classes, object categories are not described with desirable number of attributes as in supervised attribute learning.

In this thesis, we focus on attribute based image classification and propose a new approach called synthetic attributes. As the name implies, synthetic attributes are artifically generated hypothetical attributes, which are assumed to be descriptive of image classes. The proposed approach is developed for both binary and relative attributes using low level feature space representing the image classes in low level.

One of the most important advantages of synthetic attributes is that the problems originated from human factor in supervised methods are eliminated. No additional data and attribute set are used as in some binary unsupervised methods. In addition, enlarging search space problem which is occurred in unsupervised relative attribute learning is eliminated. Therefore, scalable attribute extraction allows the images to be described using desired number of the attributes in our approach.

Synthetic attributes are first implemented by randomly selecting the attributes in attribute space which contains all possible attributes representing the image classes. The size of attribute space is kept restricted with respect to the number of image classes. Therefore, the redundant attributes are discarded for compact representation. After generation of random attributes, each attribute is learned on feature space.

Random attributes are then improved by making the attribute selection more consciously. It is performed by selecting the most discriminative attributes. Attribute selection in binary representation employs the relevancy criteria and selects the most relevant attributes to the image classes. Relative attributes, on the other hand, use a ranking mechanism among the random relative attributes. Given a set of relative attributes, a rank aggregation

algorithm finds the resulting relative attribute by processing the set relative attributes jointly.

Our main goal in this thesis is to generate the attributes artificially for eliminating problems which are occurred in both supervised and unsupervised attribute based approaches and to make attribute generation independent from training dataset. We bypass the process of determining attribute and accelerate the training phase without performing any operation in feature space by our proposed approaches. Since our proposed approach is not application specific, namely, synthetic attributes are generated artificially, it can be used with different datasets which are not labeled with attributes. Synthetic attribute method is also scalable because description of object categories are represented with the desired number of attibutes.

The proposed approach has been compared to the other attribute based studies in the literature using the same data sets. The highest image classification performances obtained in other studies has been reached.

In addition, we evaluate the capacity of generalization of synthetic attributes and we perform K-shot classification. In K-shot classification some classes are left out and attributes are trained on the limited number of classes. K-Nearest Neighborhood (KNN) classifier is used for final classification. Consequently, we achieve that the results of K-shot classification are compatible with multi-class classification results in the literature.

The outline of the thesis can be given as;

Literature Review: We review mostly used methods in attribute based classification problem on the basis of supervised, semi-supervised, correlation-based and unsupervised approaches.

Random Attributes: We detail our proposed random attributes.

Attribute Selection Method: We detail our proposed attribute selection method. We will also describe the underlying structures of dependency model for Binary Attribute Selection method and some basic rank aggregation methods which are used to capture the relativity for relative attribute selection method.

Performance Analysis: We explain our datasets and experiment results.

Conclusions and Future Work: Finally, we draw a conclusion about this thesis and discuss what we could add to improve the proposed methods as a future work.

CHAPTER 2

LITERATURE REVIEW

In this chapter, we review the existing approaches proposed for attribute based image/object classification. In addition, we introduce some studies focused on only attribute learning.

We start by discussing “Supervised Approaches” in section 2.1. Afterwards, we introduce ”Correlation Based Approaches” in section 2.2 which use the relationship between object and attribute and also atribute and attribute for increasing the performance of attribute based image/object classification. We then go insight “Semi-Supervised Approaches” in section 2.3 which is developed for solving the problems originated from human factor in supervised approaches. Finally, we explain “Unsupervised Approach” in section 2.4 which eliminate the human factor in attribute learning.

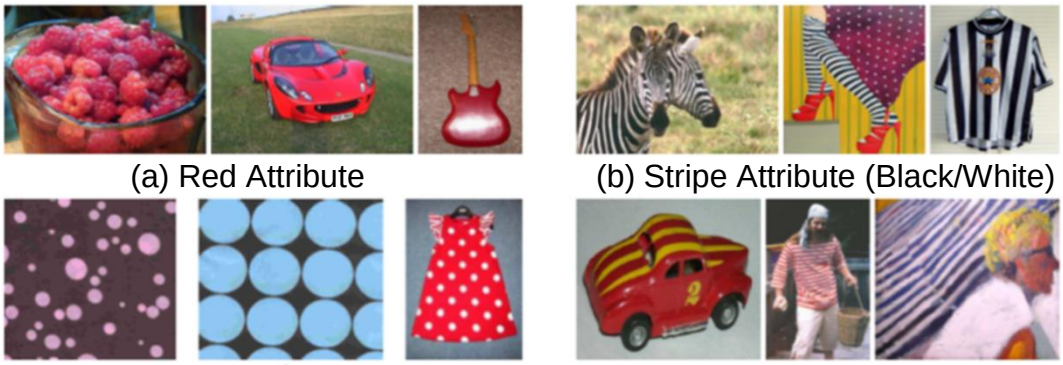
2.1 Supervised Approaches

Supervised approaches are first proposed for attribute based image/object classification and attributes are determined by human effort as binary or relative. In supervised approach, binary attribute representation is widely used in early studies [Lampert, 2009], [Farhadi, 2009], [Parikh, 2011], *etc*. A new relative attribute representation is proposed in [Parikh, 2011]. They indicate the strength of an attribute in an image with respect to other image instead of predicting the presence of an attribute (binary attribute representation). Namely, they changed the mathematical infrastructure of

attribute learning. The notion of relative attribute is highly originated from the realm of information and image retrieval.

Furthermore, visual attributes which include color, shape and texture properties of objects are first used in supervised approaches. In [Ferrari, 2007], simple color (black, white, etc.) and texture (spotted, stripped, etc.) properties of objects are used as visual attributes.

Ferrari and Zisserman focus on finding of visual attributes and it's regions it covers in a novel image. Their model process visual attributes as pattern of image segments, repeatedly sharing some characteristic properties. These patterns can be any combination of appearance, shape or layout of segments within the pattern. They use very limited dataset which contains 4 color attributes such as red, green, blue and yellow and 3 texture attributes as stripes, dots and checkerboards. Some visual attributes and image instances are shown in Figure-2.1.

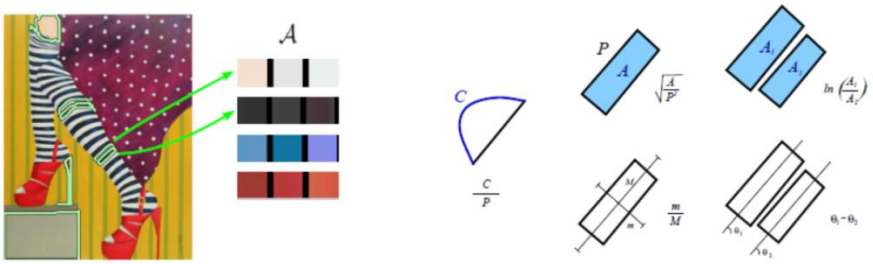


(a) Red Attribute (b) Stripe Attribute (Black/White)

(c) Round Attribute (d) Stripe Attribute (Generic)

Figure-2.1. Example images which are used to learn attributes in [Ferrari, 2007]. Two simple attributes (a and c) whose characteristic properties are captured by individual image segments (appearance for red, shape for round) and more complex attributes (b and d), whose basic element is a pair of segments are shown.

Simple attributes like “color and round” are characterized by properties of a single segment and more complex attributes like “stripes” are characterized as composed of two segments and also two segments are characterized according to their geometric properties. Geometric properties of the “stripe” attribute are shown in Figure-2.2. Various geometric proporties like curvedness, compactness and elongation, fractal dimension and area relative to image of segments are measured and sheltered for summurazing it’s shape.



(a) (b) (c) (d)

Figure-2.2. Image segments as visual features used in [Ferrari, 2007].

Furthermore, it is observed that visual attributes can not classify the object categories at an adequate classification performance. To improve the classification performance, semantic attributes which are more abstract and discriminative compared to visual attributes are used in further studies in addition to visual attributes. For example, some functional properties of objects such as “eats fish”, “walks”, “swims”, “insects”, *etc*. are used as semantic attributes in [Lampert, 2009].

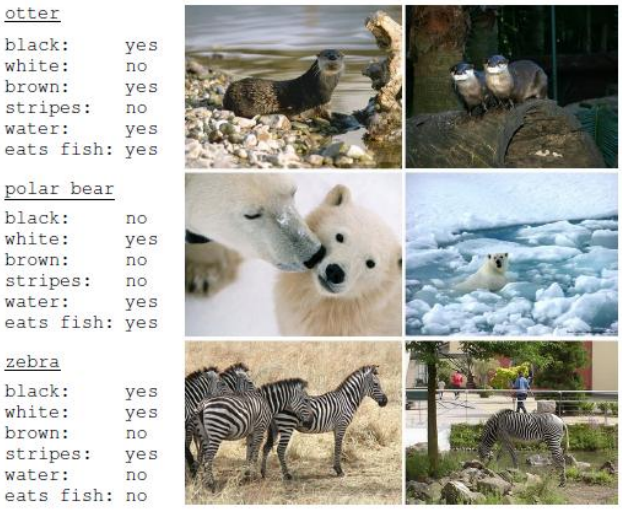


Figure-2.3. Examples of visual and semantic attributes and their training images used in [Lampert, 2009].

In addition to functional properties, some structural and hereditary properties of objects are used as visual attributes in [Farhadi, 2009].

Figure-2.4. Examples of semantic attributes and images used in [Farhadi, 2009].

Furthermore, Lampert et al focus on the problem of attribute based object classification when training and test classes are disjoint. They are originated from the fact that humans are capable of detecting completely unseen object classes when provided with a high level description. Namely, new object classes which are not seen in the training phase are detected based on the attribute based representation without the need for a new training instances.

They proposed two different methods for attribute based object classification; Direct Attribute Prediction (DAP) and Indirect Attribute Prediction (IAP) methods. The graphical representations of their methods are shown in Figure-2.5.

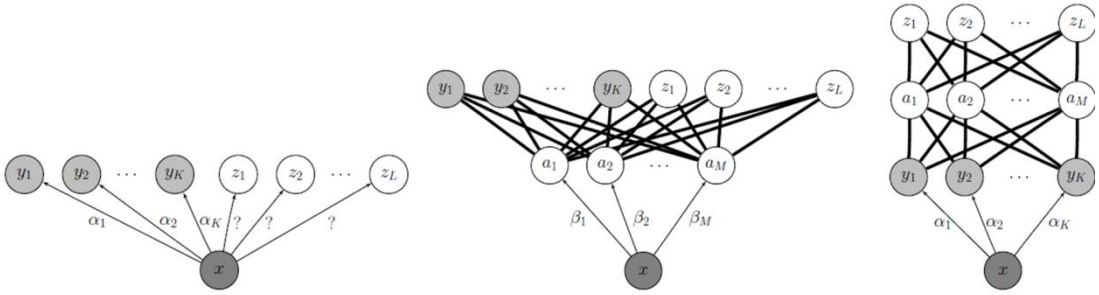


Figure-2.5. Graphical representations of methods used in [Lampert, 2009]. (a) Flat multi-class classification (classical), (b) Direct Attribute Prediction and (c) Indirect Attribute Predicition. Dark gray nodes are always observed, light gray nodes are observed only during training. White nodes are never observed but must be inferred.

In Figure-2.5, an ordinary flat multi-class classifier (a) learns one parameter α_k for each training class. It can not generalize to classes $(z_l)_{l=1\ldots L}$ that are not part of the training set. In Direct Attribute Prediction model (b) with fixed class–attribute relations (thick lines), training labels $(y_k)_{k=1\ldots K}$ imply training values for the attributes $(a_m)_{m=1\ldots M}$, from which parameters β_m are learned. At test time, attribute values can directly be inferred and these imply output class label even for previously unseen classes. In Indirect Attribute Prediction model (c), multi-class parameters α_k are learned for each training class. At test time, the posterior distribution of the training class labels induces a distribution over the labels of unseen classes by means of the class–attribute relationship.

Namely, given the situation of learning with disjoint training and test classes, if for each class z ϵ Ƶ and y ϵ Y an attribute representation a ϵ A is available, then we can learn a non-trivial classifier : X → Ƶ by transferring information between Y and Ƶ through A. We detail their proposed methods below :

1. Direct Attribute Prediction (DAP) : As shown in Figure-2.5.(b) and equation 2.1, object class and instance layers are connected via attribute layer. In training phase, only attribute classifiers a_m are learned for each different attribute according to the object class attribute labels and each attribute vector is obtained. The training of attribute classifiers and obtaining attribute parameters can be accomplished by any supervised attribute method. The trained classifiers provide the estimates of $p(a_m|x)$, from which forming a model for the complete image-attribute layer as $p(a|x) = \prod_{m=1}^{M} \mathrm{p}(a_m|x)$. At the test time, it is assumed that every class z induces it's attribute vector a^z in a deterministic way. Combining both two layer, they calculate the posterior of a test class given an image with the equation 2.1 :

$$p(z|x) = \sum_{a \in \{0,1\}^M} p(z|a)p(a|x) = \frac{p(z)}{p(a^z)} \prod_{m=1}^{M} p(a_m^z|x). \qquad (2.1)$$

After each unseen class probabality has been calculated seperately for single image, the test image is assigned to unseen object classes by using Maximum a Posteriori (MAP) estimation (equation 2.2) :

$$f(x) = \underset{l=1,\ldots,L}{\operatorname{argmax}} \prod_{m=1}^{M} \frac{p(a_m^{z_l}|x)}{p(a_m^{z_l})}. \qquad (2.2)$$

2. İndirect Attribute Prediction (IAP) : They only modify image-attribute layer unlike DAP method and they learn a probabilistic multi-class classifier estimating $p(y_k|x)$ for all training classes:

$$p(a_m|x) = \sum_{k=1}^{K} p(a_m|y_k)p(y_k|x), \qquad (2.3)$$

Furthermore, Farhadi et al focus on the determining and using the semantic attributes for shifting the goal of recognition from naming to describing. In their study, attributes are divided into three groups as parts (“has wheel”), shapes (“2D Boxy”) and metarials (“furry”) in their study. And they extract different features for each attribute group. They use color and texture for metarials, visual words for parts and edges for shapes and all features are named as “base features”.

While learning attributes, they claim that learning classifiers by fitting them to all base features often fails to generalize the semantics of attributes correctly. They support their claim with a “wheel” example. As an example; Learning a “wheel” classifier on the dataset of cars, motorbikes, buses and trains is diffucult because all examples of wheels are surrounded by “metallic” surfaces. The whell classifier might learn “metallic” instead of “wheel”. If so, when they test it on a new dataset that happens to wooden “carriage” examples, it will fail miserably, because there are not that many metalic surfaces around wheel. They focus on the accurate segmentation instead of using bounding boxes for solving the problem of across category generalization.

They use feature selection that decorrelates attribute prediction for obtaining accurate segmentation. They select features that perform well at distinguishing examples of cars with “wheels” and cars without “wheels”. By doing so, they help the classifier avoid being confused about “metallic”, as both type of examples for this “wheel” classifier have “metallic” surfaces. They select the features using regularized logistic regression [Andrew, 2004] trained for each attribute within each class and pool examples over all classes and train using the selecting features.

They extract the base features firstly and then select features that are beneficial in learning attribute classifiers. They learn attribute classifiers using selected features by Support Vector Machine (SVM) and Logistic Regression algorithms. To learn object categories, predicted attribute scores are used as features. The flowchart of their learning method is shown in Figure-2.6.

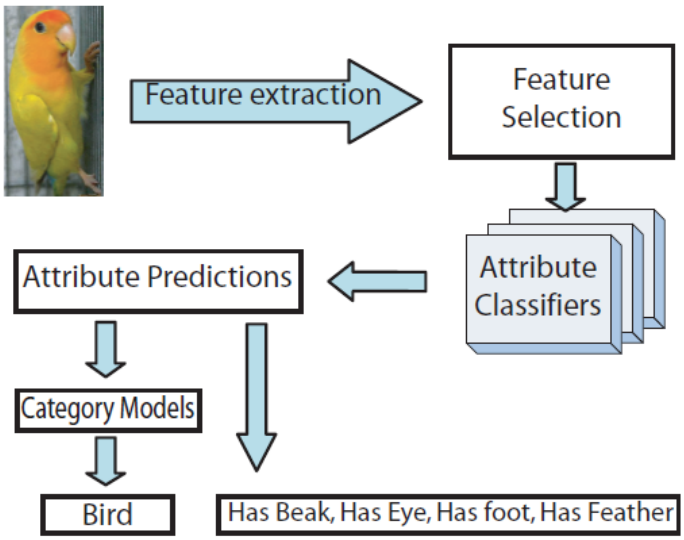


Figure-2.6. The flowchart of the learning method used in [Farhadi, 2009].

Furthermore, Kumar et al use attributes for face verification in the literature. They use 65 describable visual binary attributes such as gender, age, race, hair color, nationality, *etc.* Examples of images and attributes are

shown in Figure-2.7. They exceed the performance of classical methods used for face verification by using attributes.

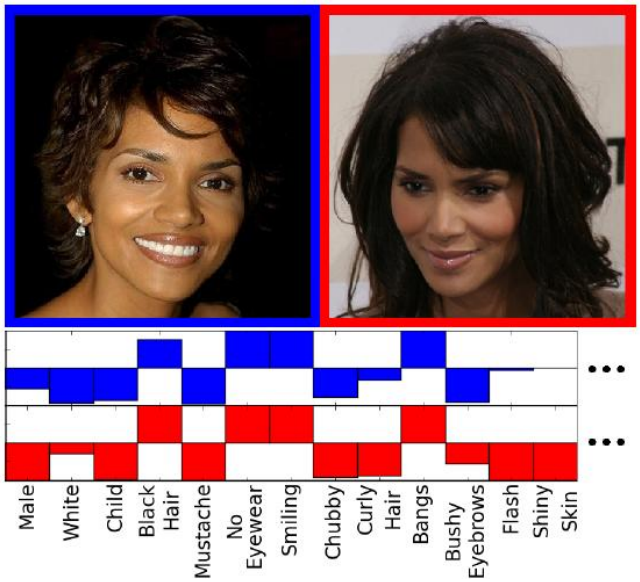


Figure-2.7. Examples of images and visual attributes used in [Kumar, 2009].

Berg et al aim to find alternative tecnique for determining attributes without hand labeled training data. Namely, their work focuses on identifiying an attribute vocabulary used to describe object categories.

They collect example images and their associated text description relating to object categories from internet (shopping sites). All associated text descriptions are considered as potential attributes. In addition, they rank their potential attributes by visualness using the learned classifiers by measuring avarage labeling precision on the validation data.

They test the effectiveness of their approach in two different cases. They first test the compatibility of some attributes which are obtained by their approach and ground truth. In this test, they get promising compatibility results for some attributes. For example, 95% average precision values for “front platform” attribute and 91% for “stiletto” attribute which are belong to

shoe object class. In second test, they compare their approach's result with human evaluation. They use the workers on Amazon Mechanical Turk (AMT) for human evaluation. Agreement between the human evaluation and their method is 70% for shoes, 80% for earrings, 80% for hanbags and 90% for ties. As a result, the attributes which are obtained by their method are matched in a high confidence with the human evaluation and ground truth text description.

Furthermore, Russakovsky and Li focus on discovering visual connections between different object classes on the basis of predetermined attributes (such as colors, shapes and textures) that are currently mising for using less image instances for learning attributes.

To construct the application specific dataset; they obtain the names of object classes from WordNet [Fellbaum, 2010] by using predetermined attributes. Furthermore, they get the image instances which are associated with names of object classes from ImageNet [Deng, 2009]. Finally, in order to obtain the ground truth data, they use workers on Amazon Mechanical Turks (AMT). Examples of object categories which are returned by [Russakovsky, 2010]'s method according to predetermined attributes are shown in Figure-2.8.

As shown in Figure-2.8, they capture the missing or hidden visual connections between very different object categories and they support the opinion that attributes are shared among different object categories. Also, this work contributes to both object classification and image retrieval. They argue that instead of using a large variety of specific object categories during training in object classification, with a little prior knowledge of attributes

(color, shape or texture) belong to specific object categories helps the classification.

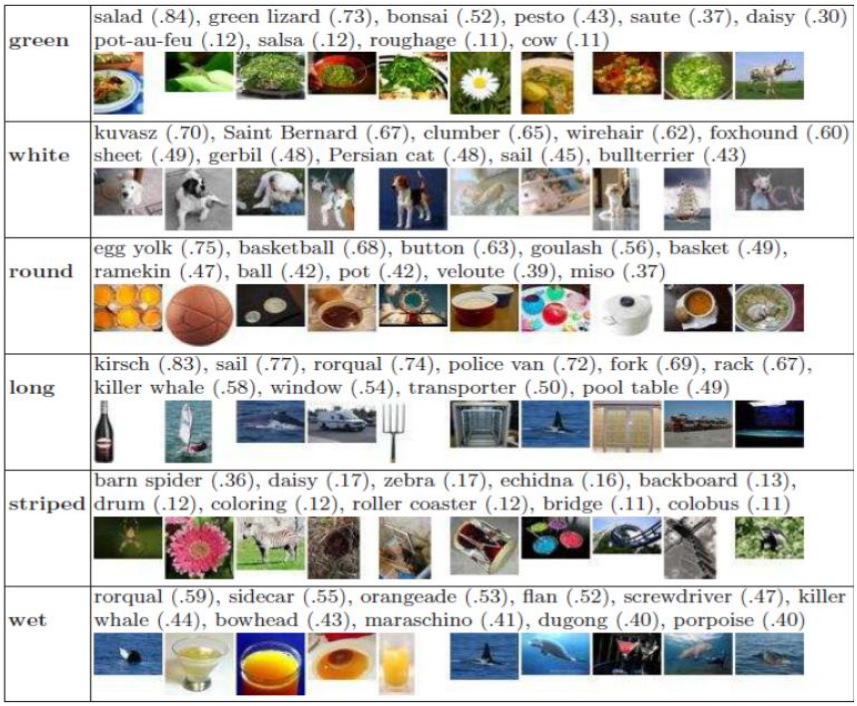


Figure-2.8. Examples of object categories which are returned by [Russakovsky, 2010]'s method according to predetermined attributes (in the leftmost rows). The top 10 object categories are showed only. The numbers in parenthesis indicates the median probability of attribute being in any object categories.

Different from supervised binary attributes, Parikh and Grauman first proposed supervised relative attributes in the literature. They claim that existing binary techniques restrict attributes to a categorical labels and thus fail to capture more general semantic relationships and are unnatural.

They evaluate their approach on two dataset. First one is Outdoor Scene Recognition (OSR) dataset [Oliva, 2001] with 8 object categories and second one is Public Figure Face (Pubfig) dataset [Kumar, 2009] with 8 object categories. They use 6 predetermined attributes for OSR and 11 predetermined attributes for Pubfig dataset. Examples of relative attributes

(the “smile” and “natural” attributes) and their training images used in [Parikh, 2011] are shown in Figure-2.9.

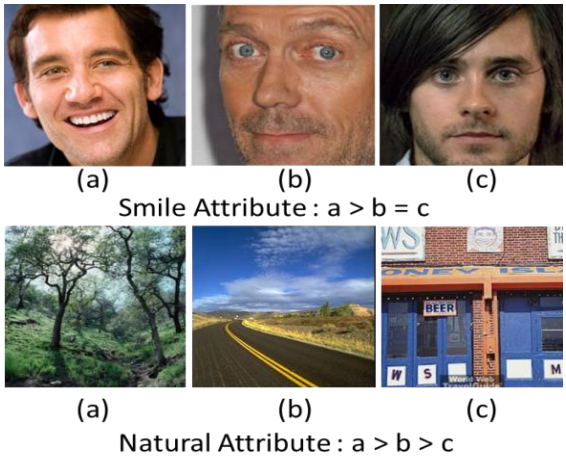


Figure-2.9. Examples of relative attributes (smile and natural) and their training images used in [Parikh, 2011]. In addition, predetermined rankings of attributes are specified according to the example images. The images which are in the leftmost rows have high rankings compared to the others according to each attributes.

In their algortihm; They represent images ($I = \{i\}$) in $\mathbb{R}^n$ by feature vector $\{x_i\}$ and a set of M attributes $A = \{a_m\}$. In addition, for each attribute a_m, they are given a set of predetermined ordered pairs of images $O_m = \{(i,j)\}$ and a set of unordered pairs $S_m = \{(i,j)\}$ such that $(i,j)\ \epsilon\ O_m \rightarrow i > j$ i.e. image i has a stronger presence of attributes a_m than j, and $(i,j)\ \epsilon\ S_m \rightarrow i \sim j$ i.e., i and j have similar strength of a_m. They learn M ranking functions:

$$r_m(x_i) = w_m^T x_i \tag{2.4}$$

For $m = 1, \ldots\ldots, M$, such that the maximum number of the following constraints is satisfied :

$$\forall (i,j) \in S_m : w_m^T x_i = w_m^T x_j \tag{2.5}$$

$$\forall (i,j) \in O_m : w_m^T x_i > w_m^T x_j \tag{2.6}$$

Finally, they solve equation 2.7 with constraints equation 2.8, 2.9 and 2.10.

$$\text{Minimize} \quad \left(\frac{1}{2}\,||w_m^T||_2^2\, C\left(\sum \xi_{ij}^2 + \sum \gamma_{ij}^2\right)\right) \tag{2.7}$$

$$w_m^T\left(x_i - x_j\right) \geq 1 - \xi_{ij}; \forall (i,j) \in O_m \tag{2.8}$$

$$w_m^T\left(x_i - x_j\right)| \leq 1 - \gamma_{ij}; \forall (i,j) \in S_m \tag{2.9}$$

$$\xi_{ij} \geq 0; \gamma_{ij} \geq 0 \tag{2.10}$$

Namely, they learn an optimization function that explicitly enforces a desired ordering on the training images. The margin is the distance between closest two projection within all desired rankings.

2.2 Correlation Based Approaches

In correlation based approaches, object-attribute and attribute-attribute relationships are used for classifiying objects or localizing objects in broad domain in an image.

Farhadi et al focus on appearance and correlation across categories for learning part and category detectors within broad domains. They put the correlation/dependency between attributes to the fore for describing objects by spatial arrangement of their attributes and the interactions between them.

They train classifiers for parts (e.g., “leg” or “wheel”), superordinate categories (e.g., “four-legged animal” or “four-wheeled vehicle”), and basic-level categories (e.g., “dog” or “car”). These classifiers model the objects as mixture of deformable “part” models. They model the objects as a mixture of two components, each with a root and five latent parts (Figure-2.10). Firstly, they obtain object candidates by accumulating votes from confident classifiers. The accumalation of voting confidence provides an initial score.

They perform inference over their graphical model to infer likelihood of the attributes. In training phase, they find all correct detections with a given confidence treshold. Then, they compute and store the off-set in scale and position (relative to scale) for each ground. This allows them to vote from parts and whole object dedectors. Namely, after finding objects by voting method, they estimate the attributes.

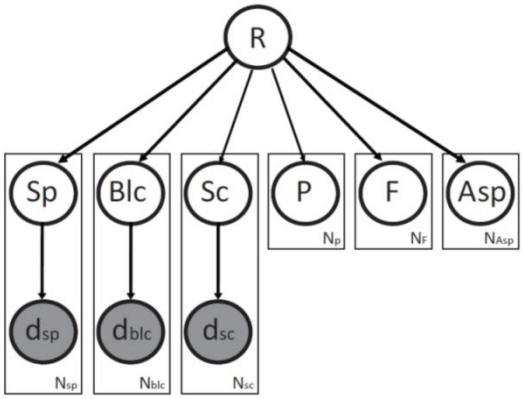


Figure-2.10. An illustration of the Graphical model used in [Farhadi, 2010].

In graphical model representation, the “root (R)” node generates each of the attributes, some of which generate detector observations. The *spatial part* (Sp) nodes encode the visibility of the parts in one of the six spatial bins (whole, top, bottom, left, center, right) in the localized object window. The *dsp* encodes the strongest detector response in each of the spatial bins. BLC stands for the *basic level categories*. The *dblc* is the maximum detector response with sufficient overlap with the region of interest. *Superordinate categories* are handled by the Sc node. Similar to the *dblc*, the *dsc* is the maximum detector response for superordinate categories.

The remaining nodes encode attributes which do not directly rely on any detector. These attributes may not be visually obvious, such as functional attributes (F), hard to predict directly, such as aspect (Asp), or not

have enough training examples to train appearance models. The node “P” indicates if an object has an attribute or not. This is different from the visibility of an attribute. For instance, “dog” has “leg” regardless of the “leg” being visible or not. For this purpose, they consider including a set of nodes “Sp” for spatial visible parts and another set of nodes “P” to consider the potentials of having a part. It also has nodes for the functional attributes of objects such as “Can this object fly?”.

Furthermore, attribute inference is done by applying equation 2.11 and 2.12. Equation 2.11 computes the marginals given the observations for attributes A_i for which learned dedectors ($A_i \epsilon \{Sp, Blc, Sc\}$). $A_j \neq A_i$ correspond to all other nodes for which they have dedectors. They find objects and their related attributes using correlation based on the spatial arrengements and the interactions between attributes.

$$P(A_i = a_i|\bar{d}) \propto \sum_R P(R)P(a_i|R)\frac{P(a_i|d_i)}{P(a_i)} * \prod_{A_{j\neq i}} \sum_{A_{j\neq i}} P(A_j|R)\frac{P(A_j|d_j)}{P(A_j)} \tag{2.11}$$

The inference on attributes $B_i \epsilon \{P, F, Asp\}$ without any learned dedectors is obtained by equation 2.12 :

$$P(B_i = b_i|\bar{d}) \propto \sum_R P(R)P(b_i|R) \prod_{A_j} \sum_{A_j} P(A_j|R)\frac{P(A_j|d_j)}{P(A_j)} \tag{2.12}$$

Furthermore, Wang and Forsyth deal with the relation between visual attributes and object classes in a different way. Their work prove that attribute and object classifiers can increase the performance of each other in

attribute classification. In training phase, they model and learn the visual attributes and object classes together. They use “joint” multiple instance learning for training the combined object class and attribute dedectors. Multiple instance learning algorithm is a kind of supervised learning and convenient for the case that only the labels of bag are known instead of the labels of instances.

Their dataset instances are labeled with object category and visual attribute label supervisedly, but not exactly location in an image. Some training instances of color-object dataset are shown in Figure-2.11.

Figure-2.11. Some training instances of color-object dataset used in [Wang, 2009]. Each row shows a color and each column shows an object class. Each color model is learned with all the images in the row and each object model is learned with all the images in the column. Color detectors and object detectors need to agree on the foreground location in intersection images.

In joint learning, T candidate windows are firstly sampled from the training image on the basis of visual saliency and homogenity. As the homogenity, they use features which are extracted in [Martin, 2004]. They

use similar method to [Moosmann, 2008] for obtaining visual saliency, and define the visual saliency as the attribute that distinguishes a concept the most from others. Each image is represented as a bag of multiple windows. Secondly, object class and visual attribute dedectors are extracted respectively from each window. Thirdly, a joint multiple instance learning algortihm is used to learn the dedectors for object and visual attributes. Lastly, the dedectors which are learned in one round are used to classify the image regions for producing a new saliency map which will be used for following rounds. The flowchart of their approach for training is shown in Figure-2.12.

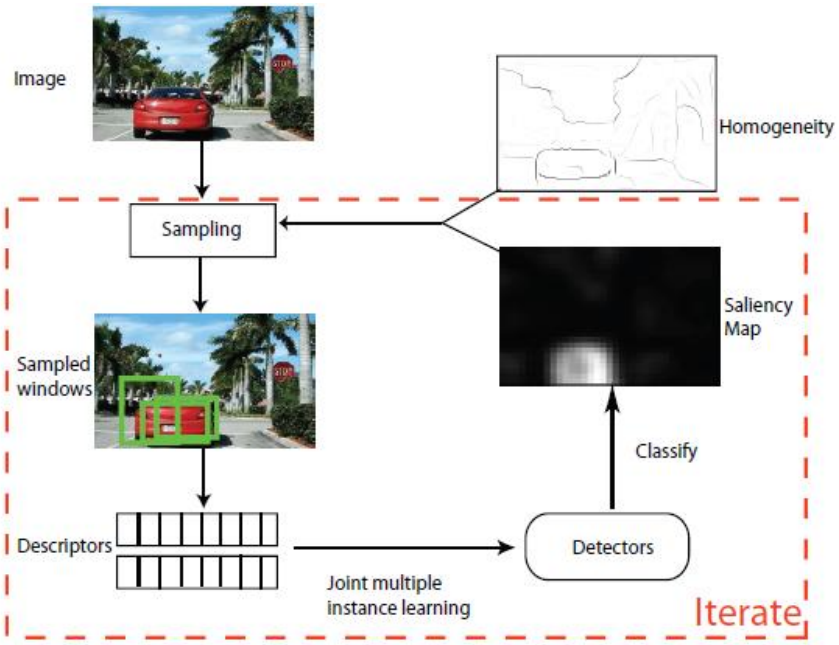


Figure-2.12. The flowchart of [Wang, 2009]'s approach for training.

Wang and Mori focus on generating a model which encapsulates the joint modeling of object class labels and their attributes. Dependency/correlations among attributes are captured by using an "undirected graphical models" which is based on supervisedly labeled

training data. They use 64 hand-labeled visual and semantic attributes such as “2D boxy”, “has hair”,“shiny”, etc.

While building the attribute relation graph, they originated from Chow-Liu algorithm [Chow, 1968] which is used for creating the dependency trees. The co-occurrence statics of attributes in the training data are used for building the attribute relation graph (Figure-2.13). In addition, normalized mutual information (NormMI) criteria is used for the amount of dependecy between pair of attributes. A strong interaction between the j-th and k-th attribute is denoted by the large NormMI and the weak interaction is vice versa. The formula of NormMI is described below :

$$NormMI(j,k) = \frac{MI(j,k)}{\min\{H(j),H(k)\}} \qquad (2.13)$$

MI = Mutual information between the j-th and k-th attribute.
H(j) = Entropy of the j-th attribute. H(k) = Entropy of the k-th attribute.

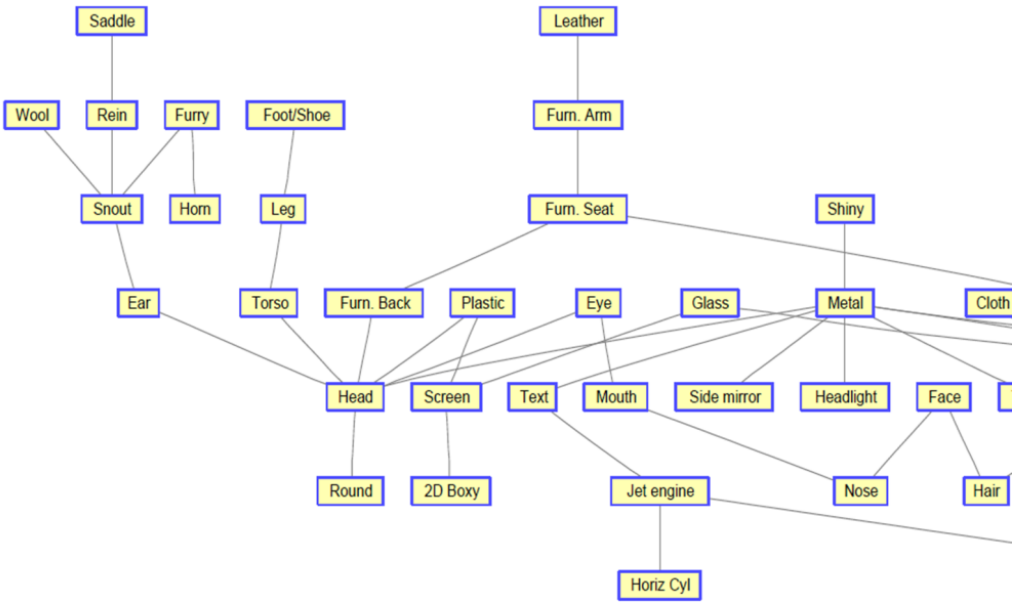


Figure-2.13. Partial visualization of the attribute relation graph learned from training data (a-Pascal dataset) [Wang, 2010].

In addition to the construction of dependency model, the utilization of dependency model scores become important detail in the whole the formula of final object classification. During the phase of final classification, dependency model scores are used to make a positive impact on the classical object class model scores for each object class. The whole formula with dependency model is decribed below :

$$h^* = argmax\, w^T \phi(x, \mathbf{h}, y) \qquad \forall\, y \in Y \tag{2.14}$$

$$w^{\mathrm{T}}(x, \mathbf{h}, y) =$$

$$w_y^T\, \phi(x) + \sum_{j \in V} w_{h_j}^{\mathrm{T}} \varphi(x) + \sum_{j \in V} w_{y,h_j}^{\mathrm{T}} \omega(x) + \sum_{j \in V} w_{(j,k)\in\varepsilon}^{\mathrm{T}} \psi(h_j, h_k) +$$

$$\sum_{j \in V} v_y, h_j \tag{2.15}$$

Object class Model $w_y^T\, \phi(x)$: This part of formula is a standart linear model for object recognition without considering any attributes.

Global attribute model $w_{h_j}^{\mathrm{T}} \varphi(x)$: This part of formula is a standart linear model trained to predict the label of j-th attribute for image x. These attributes are trained with using all object classes.

Class-specific attribute model $w_{y,h_j}^{\mathrm{T}} \omega(x)$: This part of formula is the outputs of class-specific attributes. Class-specific attributes are special for each object classes and trained on the one object class.

Attribute-attribute interaction model $w_{(j,k)\in\varepsilon}^{\mathrm{T}} \psi(h_j, h_k)$: This part of formula is the dependency model score which is obtained between j-th and k-th attributes. As an example, “foot” and “leg” attributes’s interaction will probably tend to have large values, since “foot” and “leg” attributes tend to appear in any object simultaneously.

Object-attribute interaction model v_y, h_j : This part of formula is a scalar value which indicates the strenghtness of similarity between object classes and attributes. As an example, the “people” object class and the “hair” attribute will probably have large value. Furthermore, test image is assigned to the object class which has the largest value at the final classification phase.

As a result, they perform attribute based classification using detailed attribute-attribute interaction. However, while they construct very detailed attribute relation graph for attribute-attribute interaction, they use only a scalar value based on whether an attribute is being or not in training images for object-attribute interaction which can be obtained from training data easily.

Furthermore, correlation/dependency models are frequently used in image retrieval realm in addition to attribute based object classification. Yu et al find and retrieve images using only the most relevant attributes (weak attributes) which have high dependency with query attributes instead of predetermined attributes. They refer to such attributes as “weak” because they may or may not be directly related to the query attributes and may not have clear semantic meanings. They create their weak attribute pool from hundreds or thousands of visual classifiers which are not directly related with their dataset and query attributes. In testing phase, when one or more query attributes are given for image retrieval, they choose the most related weak attributes with query attribute on the basis of mutual information from the pool of weak attributes. In addition, they change the number of weak attributes for different datasets very easily due to the highest mutual information values.

Two notions attract our attention in [Yu, 2012]. First one is that weak attributes are obtained from different studies and they are not directly related with their dataset and query attributes. Owing to this study, we conclude that the notion of “how much the weak attributes support the query attributes for final classification” is more important than the way of creating attributes. And the second one is that they chance the number of weak attribute for different datasets**.**

2.3 Semi-Supervised Approaches

In semi-supervised approaches, attribute based image/object classification performance is increased by using only unsupervised binary attributes in addition to supervised attributes.

Kumar et al try to remove the manual labeling of attributes in their “smile classifiers” methods in additon to classical binary attributes for face verification. The basic idea for smile classifier method is that they describe a person’s appearance in terms of the similarity of different parts of their face to a limited set of “reference” people. Each smile classifier is trained using several images of a specific reference person limited to a small face region such as the eyes, nose or mouth (Figure-2.14). Instead of labeling attributes by hand, they label the reference people and their image regions by hand and they argue that the labeling process of reference people for constructing smile classifiers is simpler than the labeling of attributes for attribute based image/object classification. Finally, they represent the images based on the reference people (Figure-2.15).

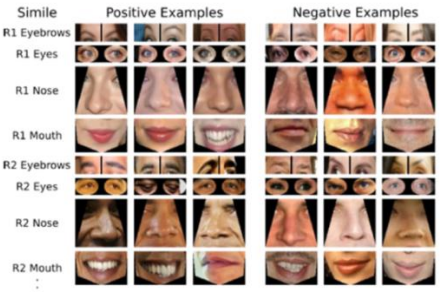


Figure-2.14. Example image patches for constructing smile classifiers [Kumar, 2009]. Positive and negative examples based on four regions (eye, mouth, eyebrows and nose) of reference (R1 and R2).

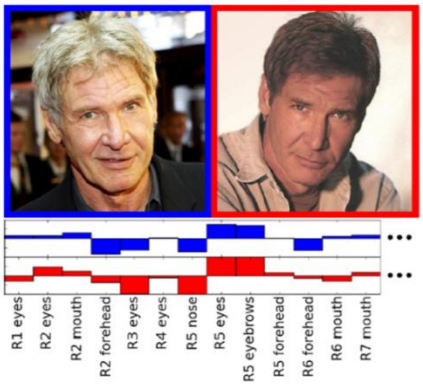


Figure-2.15. Image representation relative to the reference (R1 and R2) people [Kumar, 2009].

Furthermore, Farhadi et al proposed discriminative attributes in addition to semantic binary attributes for distinguishing the classes which have nearly the same semantic attributes. The auxiliary discriminative attributes take the form of random comparisons. In each comparison, a portion of data is splited into two partitions. Instances not belonging to the selected classes or attributes are not considered. They create the discriminative attributes for only the classes which are not discriminated by predetermined visual or semantic attributes. They use a linear SVM to learn tens of thousands of these splits and pick those that can be well predicted using the validation data. Finally, 1000 auxiliary discriminative attributes are

selected by human effort among the huge number of learned auxiliary discriminative attributes.

2.4 Unsupervised Approach

In unsupervised approach, we found only one study. Ma et al use fully unsupervised relative attributes to eliminate the problems which are originated from human factor. They formulate the attribute learning as a mixed-inter programming problem and they propose an algorithm to solve it approximately. They use Outdoor Scene Recognition (OSR) and Public Figure Face (Pubfig) datasets which are the same datasets used in [Parikh, 2011].

Their main formulation is based on the one used in [Parikh, 2011]. However, they only consider relative attributes that contain strict pairwise order (a>b or a<b) and ignore the equality situation of classes (a=b). In order to create different useful attributes from each other, they use pairs of different classes at the start of creating attributes. Their key idea is that choosing a broad set of initializations that tend to yield a broad set of useful attributes.

As shown in Figure-2.16, they determine the initial ranking of two classes randomly and create ranking fuction/weight vector by using 2 classes in the start of the generation of an attribute. Folowing by , they add the other remaining classes to feature space on the basis of entropy criteria. For adding new classes to feature space, after generating the initial weight vector by using 2 classes, they first calculate the mean points of classes on the weight vector by projecting each instances of classes to line. Secondly, they divided the line (weight vector) into bins on the basis of these points. Subsequently, they compute the entropy for each remaining classes

according to bins and they select the class that has smallest entropy. In addition, they determine relative ranking of classes according to their means of classes among each other. Finally, they update the weight vector by using 3 classes. This process is continued until the last class to be added into feature space by using the same procedures. The algorithm is run multiple times initialized with every possible pair of classes and each run yields a candidate relative attribute.

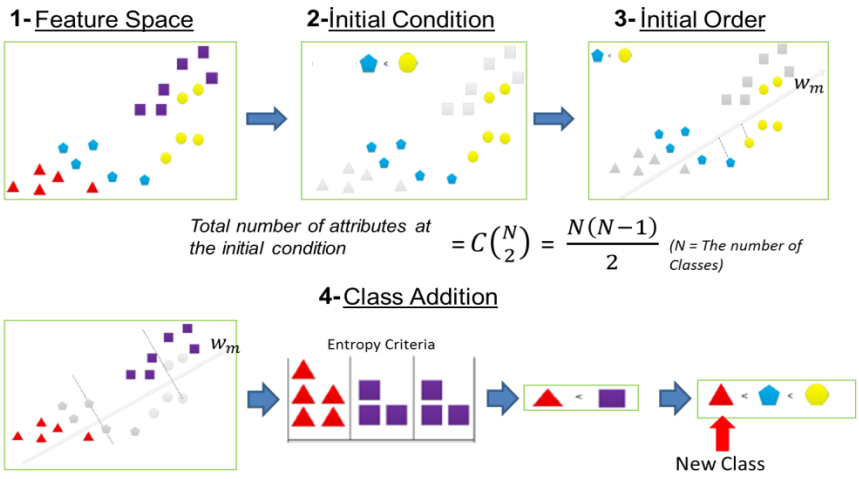


Figure-2.16. Overview of creating single unsupervised relative attribute [Ma, 2012]. There are 4 classes in feature space.

As a result, they exceed the classification performance of [Parikh, 2011] on the same dataset with the same setting by using unsupervised relative attributes. We evalute that since they can create maximum number of different relative attributes up to 2-combination of N ($\binom{N}{2}; N = number\ of\ image\ classes$), attribute scalability problem is still continued like supervised aproaches and many operations are performed in the feature space. In addition, there is no corresponding semantic meaning of unsupervised relative attribute which is extracted on the basis of entropy criteria in an image.

CHAPTER 3

RANDOM ATTRIBUTES

In this chapter, we explain the disadvantages of current attribute based approaches in section 3.1 and introduce our binary and relative random attribute approaches in section 3.2 and 3.3 respectively.

3.1 Problems in Current Attribute Based Approaches

In supervised approaches, since attributes are determined by humans, image classes may not be represented with the desired number of attributes. Although human can determine general properties of object classes in an adequate level, they have difficulties in determining discriminative properties of objects. Since attribute generation requires too much effort, the number of supervised attributes are not increased easily. In addition, supervised attributes are also created for only specific datasets. If we change the dataset, attribute generation determination process must be repeated for new dataset. Therefore, the supervised attribute generation process is time consuming.

In semi-supervised approaches, supplementary datasets or processes are used for creating unsupervised binary attributes. In [Kumar, 2009], Pubfig dataset is used for only creating unsupervised binary atrributes. In [Farhadi, 2009], tens of thousands of auxiliary discriminative attributes are only created for discriminating the only object classes which have nearly the same attributes and auxiliary discriminative attributes are selected among them by

human effort. This approach is also effort and time consuming as in supervised approaches.

In unsupervised relative approach, many operations are performed in feature space to obtain the discriminative relative attributes. To obtain different attributes, they choose different pair of image classes and this sutiation limit the number of relative attributes. For example, in the case of k different image classes in dataset, different relative attributes can be created up to 2-combination of k ($\binom{k}{2}$). Furthermore, when the number of classes increases, the excessive growth of the search space appears to be a major problem in feature space. In addition, suppose that the attribute extraction operation is made over image instances instead of image classes, the enlarging problem in the search space would be unavoidable.

3.2 Binary Random Attributes

In the creation phase of classical binary attributes, all object classes are divided into two group and labeled as 0 or 1 manually for each different attribute. In other words, feature space is divided into two group using the instances of object classes for one binary attribute. Furthermore, feature space is divided into various subspaces to create all attributes. Finally, an image attribute vector is created by determining which subspaces that the feature vector belong to. The creation of one classical binary attribute is shown in Figure-3.1.

In binary random attribute approach, hypothetical attributes are extracted instead of predetermined attributes. During the creation of a hypothetical attribute, all image classes are randomly divided into two group

and each image is assigned to one of two groups. The current attribute is then represented by one of them. In our approach attributes are considered as a joint hypothetical property which is shared by some image classes.

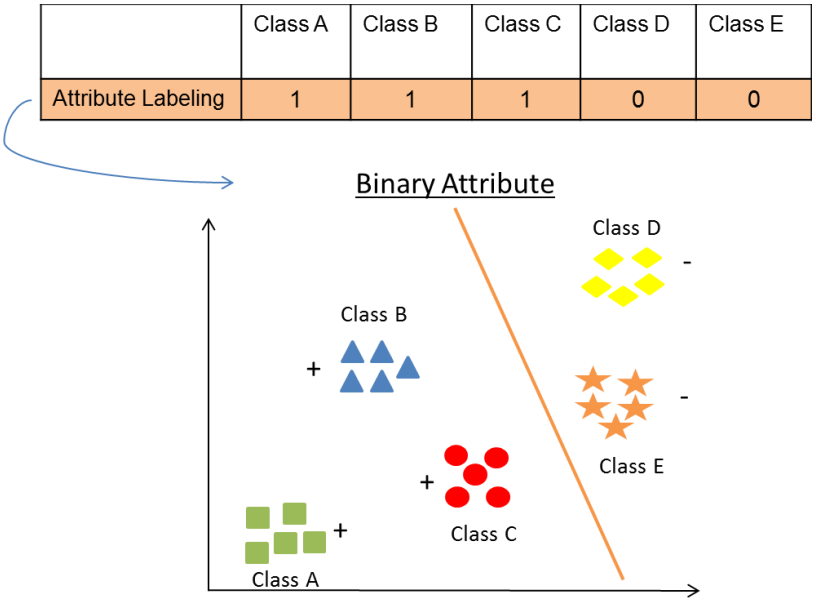


Figure-3.1. The creation of one classical predetermined binary attribute. There are 5 classes in this example.

When the image classes are randomly grouped, feature space is divided into two random subspaces. One of those two subspaces corresponds to the presence of a hypothetical attribute. It is assumed that the image instances inside the subspace have that attribute. All instances are then labeled as 1 for the corresponding attribute. On the other hand, the other subspace represents the absence of that attribute. All the other instances in this subspace are labeled as 0. The decision which subspace corresponds to the presence or absence of an attribute is made randomly.

After the labeling of each hypothetical attribute is performed using the above procedure, an image is described by a binary attribute vector. Each

element of this vector represents the presence or absence of the corresponding hypothetical attribute in the image.

If there are k image classes in dataset, we can create different binary attributes up to the 2 raised to the power of k. $(2^k - 2; k = the\ number\ of\ classes)$. Therefore, we classify the object classes with desired number of random binary attributes.

We show disadvantages of supervised binary attribute approach in Table-3.1 which contains the attribute labelings in two public dataset. In those datasets, to reduce the human effort, attribute assignments are performed on object classes instead of each images. Because of this rough labeling, different attiributes may represent the same or similar properties of image classes. Since those attributes are represented in the same similar labels, their discriminative power diminishes. Therefore, this situation leads to poor performance of the image/object classification**.**

As shown in Table-3.1, although “perspective” and “diagonal-plane” attributes in OSR dataset have different meanings in reality, they basically represent the same property in the attribute space. Similarly, the attributes of “Big-Lips” and “Round-face” are the same property in Pubfig dataset. They use the same instances of object classes as positive (1) and negative (0) examples during attribute training. Namely, feature space is divided into the same subspaces for different attributes.

In binary random attribute approach, we minimize the attribute similarity and maximize the classification performance very easily with the help of obtaining attributes in a hypothetical space. Consequently, training

phase of random attributes is conducted more quickly than both existed supervised and unsupervised attribute methods.

Table-3.1. Supervised binary attribute assignments used in [Parikh, 2011]. The upper side is belong to OSR dataset and the down side is belong to Pubfig dataset. Left column shows attribute names and rigth column shows binary attribute assignments. T, I, S, H, C, O, M, and F letters represent the 8 different object classes.

	Binary
OSR	T I S HC OMF
natural	0 0 0 0 1 1 1 1
open	0 0 0 1 1 1 1 0
perspective	1 1 1 1 0 0 0 0
large-objects	1 1 1 0 0 0 0 0
diagonal-plane	1 1 1 1 0 0 0 0
close-depth	1 1 1 1 0 0 0 1
PubFig	ACHJ MS V Z
Masculine-looking	1 1 1 1 0 0 1 1
White	0 1 1 1 1 1 1 1
Young	0 0 0 0 1 1 0 1
Smiling	1 1 1 0 1 1 0 1
Chubby	1 0 0 0 0 0 0 0
Visible-forehead	1 1 1 0 1 1 1 0
Bushy-eyebrows	0 1 0 1 0 0 0 0
Narrow-eyes	0 1 1 0 0 0 1 1
Pointy-nose	0 0 1 0 0 0 0 1
Big-lips	1 0 0 0 1 1 0 0
Round-face	1 0 0 0 1 1 0 0

3.3 Relative Random Attributes

In relative random attribute approach; it is aimed to sort the image samples with respect to the corresponding attribute. In order to achieve this goal, image instances are projected into a line in the feature space. The creation of one relative attribute is shown in Figure-3.2.

In random relative attribute approach, the ranking of image classes is given randomly in the training phase. In a similar manner, we solve an optimization problem for determining the parameters of ranking function which builds the projection line. Namely, training phase is accomplished by

ranking the hypothetical relative attributes randomly and the strength of an attribute in an image is coded with respect to other images.

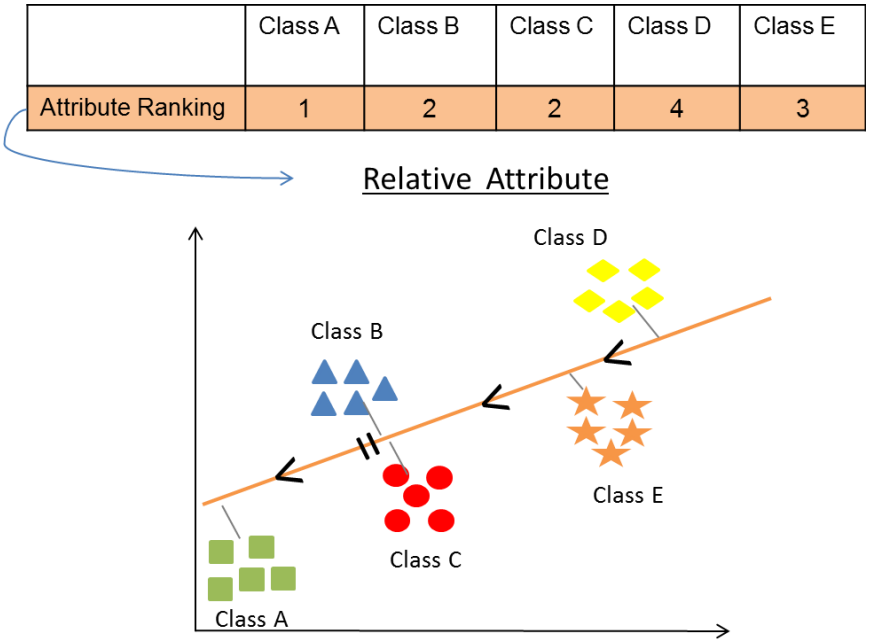


Figure-3.2. The creation phase of one relative attribute. There are 5 classes in this example.

In addition, the number of relative hypothetical attributes can be increased to the number of the all possible sequence which consist of equality (a=b) or greatness (a>b) of object classes, as in the binary random attributes. Namely, If there are k image classes in dataset, we can create $k!$ different relative hypothetical attributes.

CHAPTER 4

ATTRIBUTE SELECTION

In this chapter, we first explain “Binary Attribute Selection Method” in section 4.1. We introduce underlying structure of our dependecy model with “Entropy, Joint Entropy, Conditional Entropy and Mutual Information” in section 4.1.1 and “Maximum Relevance and Minimum Redundancy Method (mRMR)” in section 4.1.2. Then, we describe our “Proposed Binary Method” in section 4.1.3 on the basis of dependency model. Followed by, “Relative Attribute Selection Method” is explained in section 4.2. In section 4.2, we first describe the “rank aggregation” in section 4.2.1 and its “Borda” and “Kemeny-Young” methods which are used to capture the relativity for relative ranking assignments in section 4.2.1.1 and 4.2.1.2 respectively. Finally, “Proposed Relative Method” is introduced in section 4.2.2.

In attribute selection approach; we make the attribute selection more consciously compared to random attributes. Namely, we focus on finding most relevant and discriminative attributes for object classes and put the relation between object and attributes to the fore. We evaluate that the “has eyes” attribute and the “has nose” attributes complement each other and provide prior probability for each other. For example, when we detect the “has eye” attribute correctly in an image, “has mouth” attribute must be presented in the same image except unordinary conditions. We build the relation arthitecture on the basis of maximum relavance and minimum redundancy (mRMR) method. We obtain the most relevant and discriminative attributes for object categories by mRMR method and combine them for final

classification. To create the relative synthetic attributes, we use some basic rank aggregation methods. Consequently, we perform final classification with the most relevant and discriminative attributes of object classes.

4.1 Binary Attribute Selection Method

4.1.1. Entropy, Joint Entropy, Conditional Entropy and Mutual Information (MI)

Entropy is a measure of the uncertainty in a random variable. The concept was introduced by Claude E. Shannon in his 1948 paper "A Mathematical Theory of Communication".

Let X be a discrete random variable and $p(x)$ is a probabality mass function; and the entropy is formulated with equation 4.1 :

$$H(X) = -\sum_{x \in X} p\,(x) \log p\,(x) \tag{4.1}$$

Note that entropy is a functional of the distribution of X. It does not depend on the actual values taken by the random variable X, but only on the probabilities. Since the entropy measures the uncertainty in X, we can say that $H(X)$ is approximately equal to how much information we learn on average from one instance of the random variable X.

After the definition of entropy of a single random variable, even if we have a pair of random variables (X,Y), we must explain the joint and conditional entropy. Actually, there is nothing new, because (X,Y) can be considered to be a single vector-valued random variable. Joint entropy is the entropy of a joint probability distribution or a multi-valued random variable. The joint entropy measures how much uncertainty there is in the two random

variables X and Y taken together. The joint entropy $H(X,Y)$ of a pair of discrete random variables (X,Y) with a joint distribution $p(x,y)$ is defined as ;

$$H(X,Y) = -\sum_{x \in X}\sum_{y \epsilon Y} p\,(x,y)\log p\,(x,y) \qquad (4.2)$$

In addition, the conditional entropy is a measure of how much uncertainty remains about the random variable Y when we know the value of X. If (X,Y)~ $p(x,y)$, then the conditional entropy $H(Y|X)$ is defined as;

$$H(Y|X) = -\sum_{x \epsilon X} p\,(x)\,\mathrm{H(Y|X = x)} \qquad (4.3)$$

$$= -\sum_{x \in X} p(x) \sum_{y \epsilon Y} p\,(y|x)\log p\,(y|x)$$

The naturalness of the definition of joint and conditional entropy is exhibited by the fact that the entropy of a pair of random variables is the entropy of one plus the conditional entropy of the other (chain rule-equation 4.4).

$$H(X,Y) = -\sum_{x \in X}\sum_{y \epsilon Y} p\,(x,y)\log p\,(x,y) \qquad (4.4)$$

$$= -\sum_{x \in X}\sum_{y \epsilon Y} p\,(x,y)\log \mathrm{p(x)}\,p\,(y|x)$$

$$= -\sum_{x \in X}\sum_{y \epsilon Y} p\,(x,y)\log p(x) - \sum_{x \in X}\sum_{y \epsilon Y} p\,(x,y)\log\, p\,(y|x)$$

$$= -\sum_{x \epsilon X} p\,(x)\log p(x) - \sum_{x \in X}\sum_{y \epsilon Y} p\,(x,y)\log\, p\,(y|x)$$

$$H(X,Y) = H(X) + H(Y|X)$$

Within the framework of the previously mentioned issues, mutual information is a quantity that measures a relationship between two random

variables that are sampled simultaneously. In particular, it measures how much information is communicated, on average, in one random variable about another. It is the reduction in the uncertainty of one random variable due to the knowledge of the other. We can formulate the definition of mutual information $I(X;Y)$ as;

$$I(X;Y) = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \tag{4.5}$$

$$= \sum_{x,y} p(x,y) \log \frac{p(x|y)}{p(x)p(y)}$$

$$= -\sum_{x,y} p(x,y) \log p(x) + \sum_{x,y} p(x,y) \log p(x|y)$$

$$= -\sum_{x,y} p(x,y) \log p(x) - \left(-\sum_{x,y} p(x,y) \log p(x|y) \right)$$

$$I(X;Y) = H(X) - H(X|Y)$$

Thus, the mutual information $I(X;Y)$ is the reduction in the uncertainty of X due to the knowledge of Y. By symmetry, it also follows that;

$$I(X;Y) = H(Y) - H(Y|X) \tag{4.6}$$

Since $H(X,Y) = H(X) - H(Y|X)$ as shown in equation 4.6, we can also formulate mutual information as below;

$$I(X;Y) = H(X) + H(Y) - H(X,Y) \tag{4.7}$$

According to equation 4.7, we can show the relationship between mutual information and entropy in Figure-4.1.

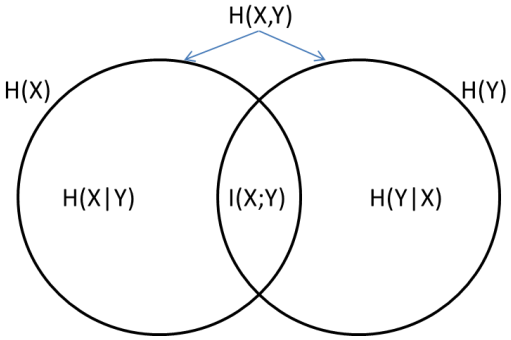


Figure-4.1. Relationship between mutual information and entropy.

4.1.2. Maximum Relevance and Minimum Redundancy (mRMR)

The mRMR method is firtly proposed in [Peng, 2005] to select good features according to the maximal statistical dependency and minimal redundancy criterion based on mutual information. Because of the difficulty in directly implementing the maximal dependency condition in feature selection, they perform an equivalent form, called mRMR for the feature selection.

In feature selection, it is demonstrated that the combinations of individually good features do not necessarily lead to good classification performance [Webb, 1999], [Cover, 1991], [Cover, 1974], [Jain, 2000], [Peng, 2005]. Because of this reason, some researchers have studied to reduce the redundancy among features and select features with minimal redundancy [Kwak, 2002], [Peng, 1997], [Kohavi, 1997], [Li, 2000], [Cover, 1974], [Jain, 2000]. Consequently, in [Peng, 2005], they combine max-relevance and min-redundacy different from other mentioned studies.

In [Peng, 2005], max relevance criterion is used as similar to max-dependency. The criteria of max-relevance is to search features which approximates $D\ (S, c)$ (The dependency between a feature set S and class c)

with the mean value of all mutual information values between individual feature x_i and class c :

$$\max \mathrm{D}\,(S,c),\ D = \frac{1}{|S|}\sum_{x_i \in S} I\,(x_i;c) \tag{4.8}$$

Since the features which are selected by max-relevance criterion have rich redundacy, they perform minimal-redundancy criterion in addition to max-relevance. They claim that when two features are highly dependent on each other, the respective class discriminative power would not change much if one of them removed. Therefore, they use minimal redundancy criterion to select mutually exclusive features;

$$\min \mathrm{R}\,(S), \qquad r = \frac{1}{|S|^2}\sum_{x_i, x_j \in S} I\,(x_i;x_j) \tag{4.9}$$

Furthermore, they define the operator $\Phi\,(D,R)$ to combine D and R and consider the following simplest form to optimize D and R simultaneously;

$$\max \quad \Phi\,(D,R), \Phi = D - R \tag{4.10}$$

In practice, they use incremental search methods (equation 4.11) to find the optimal features defined by $\Phi\,(.)$. In detail, since they already have S_{m-1}, the feature set with $m-1$ features, they select the $m\,th$ feature from the set $\{X - S_{m-1}\}$. Finally, they select the feature which maximizes $\Phi\,(.)$ and add it to S iteratively;

$$\max_{x_j \,\epsilon\, X - S_{m-1}} \left[I(x_j;c) - \frac{1}{m-1}\sum_{x_i \epsilon\, S_{m-1}} I(x_j;x_i)\right] \tag{4.11}$$

4.1.3. Proposed Binary Method

In binary attribute selection method, we first determine the class-specific attributes, each of those attributes is unique to one of the image

classes. In other words, class specific attribute is assumed to exist only in that class. Furthermore, we create dependecy model to obtain the most related attributes with class-specific attributes on the basis of mRMR method. Our binary attribute selection method is summarized in Figure-4.2.

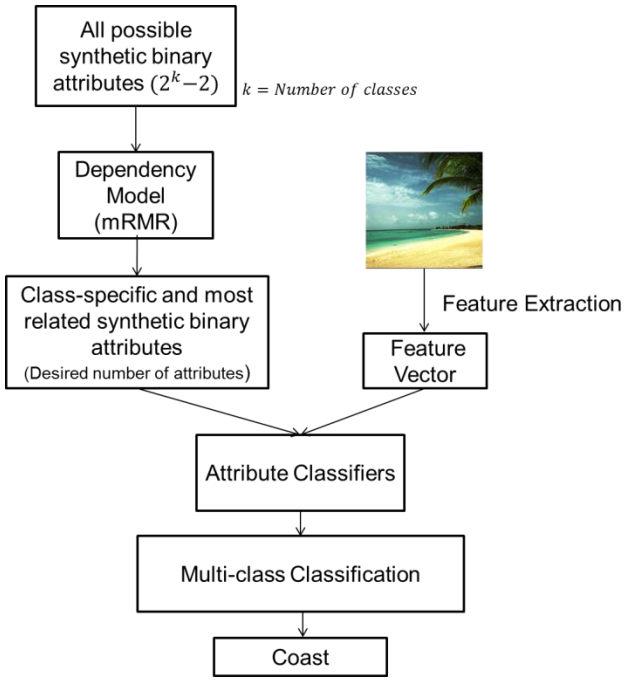


Figure-4.2. The flowchart of our proposed binary attribute selection method. k represent the number of classes.

Feature space can be divided into maximum $2^k - 2$ different subspaces, if the dataset contains k different classes. Namely, we can obtain maximum $2^k - 2$ hypothetical binary attributes different from each other. Considering all possible attributes, each of binary attributes has different degree of importance for the identification of different object classes.

Generally, people can distinguish different objects even a single individual characteristic of object is available. Therefore, we primarily search the attributes which are specific for each object classes. We determined the attributes which are presence in one class and absence in all other remaning

object classes as class-specific attributes. So, we represent one object class with one attribute and obtain the class-specific attributes as much as the number of object classes. The labeling of class-specific attributes is shown in Table-4.1. Consequently, we use the class-specific attributes as the first element of our method.

Table-4.1. Illustration of class-spesific attributes according to each object class. Class-specific attributes are indicated in columns for different object classes. The attribute which are presence in one class and absence in all other remaning classes are considered as “class-specific attribute”. In this table, there are 4 different object classes (A,B,C and D) and 4 class-specific attributes.

		Specific Attribute for class A	Specific Attribute for class B	Specific Attribute for class C	Specific Attribute for class D
Object Classes	A	1	0	0	0
	B	0	1	0	0
	C	0	0	1	0
	D	0	0	0	1

Furthermore, we relate the remaining attributes $[(2^k - 2) - class\ specific\ attributes]$ with class-specific attribute by using mRMR method for each object class seperately. After implementing dependency model, we faced with the attributes which is specific for two classes as the most related attributes with class-specific attributes (Table-4.2).

Table-4.2. İllustration of two class-specific attributes. The specific attributes for 2 classes are indicated in columns. In this table, there are 4 different object classes (A,B,C and D) and 4 two class-specific attributes.

		Specific for classes A and B	Specific for classes B and C	Specific for classes A and C	Specific for classes B and D
Object Classes	A	1	0	1	0
	B	1	1	0	1
	C	0	1	1	0
	D	0	0	0	1

After obtaining the class-specific and their related attributes, we combine them together. Since we built dependency model for each object class separately, we found some overlapped attributes in desired attribute dimension. The conflicting situations are; 1) The attributes which have the same importance for different object classes or 2) The attributes which is more important for one object class than the other object class. When we encounter with overlapped attributes, we don't use the same attributes twice in final attribute set and engage the second most important attribute for that object class.

We explain our combining method with Table-4.3. It can be seen that 7^{th} attribute has the same importance for class B and C and 5^{th} attribute is more important for class A than class B. In these situations, we don't use these attributes twice. One can say that sequence of classes may change the whole final attribute set. For example, suppose we want to obtain 10 attributes by using the sequence of class A, B, C, D, attribute set will be formed as; 1, 2, 3, 4, 5, 7, 14, 8, 11, 12. Even if we use the sequence of class

D, C, B, A, whole attribute set will be formed as 4, 3, 2, 1, 14, 7, 5, 12, 11, 12. Since we only pay attention to the content of whole final attribute set instead of attribute sequence, there is no difference in content of whole final attribute set when we change the sequence of classes. The selection of sequence of classes makes little impact on changing of content of attribute set in the case of large dataset with huge number of attributes. We evaluate that this difference is insignificant.

Table-4.3. Dependency table according to each object class. This table shows the sequence of attribute relation with object classes. In this example, there are 4 classes and 14 ($(2^4 - 2)$) attributes. 1, 2, 3 and 4th atttributes are class-specific attributes. Each row indicate the ranked attributes (relevant attributes) according to the different classes. Each column indicates the degree of importance of attribute for each class. For example, 5th attribute is the most related attribute for class A and it is the least related attribute for class D.

	The ranking of relationship for each object class.										
Object Classes		1	2	3	4	5	6	7	8	9	10
	A (1)	5	8	10	14	13	7	6	9	11	12
	B (2)	7	8	5	6	10	9	11	14	12	13
	C (3)	7	11	14	5	6	13	8	12	9	13
	D (4)	14	12	11	8	10	13	8	7	6	5

Consequently, we find all class-specific and their most related attributes for attribute based object classification. The most important aspect of our dependency model is that it is built on the basis of only hypothetical attributes before attribute training process.

4.2 Relative Attribute Selection Method

4.2.1 Rank Aggregation

The rank aggregation problem is to combine many different rank orderings on the same set of candidates or alternatives in order to obtain a “better” ordering. Rank aggregation has been studied extensively in the context of social choice theory and is a classical problem which deals with voting and so on [Borda, 1781], [Condorcet, 1785]. The problem also arises in many other settings: Sports and Competition: How to determine the winner of a season, how to rank players or how to compare players from different eras?, Machine Learning: Collaborative filtering and meta-search, Statistics: Notions of Correlation, Database Middleware: Combining results from multiple databases [Dwork, 2001]. The problem of rank aggregation has become very popular due to developments of information retrieval on the internet, online shopping stores, recommendation systems and so on [Yasutake, 2012] .

4.2.1.1. Borda’s Method

Borda’s method is firstly proposed in [Borda, 1781] for elections and than used in the problem of rank aggregation. It ranks the candidates due to the voter’s preference. In the phase of ranking, it assigns a weight/score corresponding to the positions in which a candidate appears within each voter’s ranked list. After assigning weights to all candidates, candidates are sorted by their total score. An example of the calculation of Borda’s method is presented in Figure-4.3. The Borda’s method is used for rank fusion problem in mostly recent studies [Dwork, 2001], [Aslam, 2001] and [Renda, 2003].

	Voter preference-1	Voter preference-2	Voter preference-3	Voter preference-4
1	A	D	D	A
2	B	A	B	D
3	C	C	A	B
4	D	B	C	C

A = 4 + 3 + 2 + 4 = 13;
B = 3 + 1 + 3 + 2 = 9;
C = 2 + 2 + 1 + 1 = 6;
D = 1 + 4 + 4 + 3 = 12;

Borda Winner is A;
All ranking is A > D > B > C.

Figure-4.3. An example of the calculation of Borda's method. It is presented due to the four voter's preferences.

Improtant advantages of Borda's method are that it is computationally very simple and can be implemented in linear time [Dwork, 2001]. It also supply the properties called consistency, anonymity and neutrality in the social choice literature [Young, 1974].

4.2.1.2. Kemeny-Young Method

Kemeny-Young method is firstly proposed in [Kemeny, 1959] for elections. In this method, pairwise comparison counts to identify the most popular choices in an election. This method assigns a score for each possible sequence, where each sequence considers which choice might be most popular, which choice might be second-most popular, which choice might be third-most popular, and so on down to which choice might be least-popular. The sequence that has the highest score become the winning sequence.

Kemeny–Young calculations are usually done in two steps. The first step is to create a matrix or table that counts pairwise voter preferences (Table-4.4). The second step is to test all possible rankings, calculate a score for each such ranking and compare the scores (Table-4.5). The winner ranking is choosen due to the final scores. Even if more than one ranking has the same largest score, all these possible rankings are tied and typically the overall ranking involves one or more ties.

Table-4.4. The first step of Kemeny-Young Method based on the 100 voter's preferences. If we detail last column, C is preferred over D by 83 voter, vice versa by 17 voters.

All possible pairs of choose name	Number of votes with indicated preference	
	Prefer X over Y	Prefer Y over X
X= A Y= B	42	58
X= A Y= C	42	58
X= A Y= D	42	58
X= B Y= C	68	32
X= B Y= D	68	32
X= C Y= D	83	17

Table-4.5. The second step of Kemeny-Young method. This table contains all possible rankings of A, B, C and D candidates. The winner ranking is B>C>D>A with the higgest ranking score 393.

First Choice	Second Choice	Third Choice	Fourth Choice	Raking Score
A	B	C	D	345
…	…	…	…	…
B	***C***	***D***	***A***	***393***
…	…	…	…	…
D	C	B	A	255

4.2.2. Proposed Relative Method

Relative Attribute Selection method is developed on the basis of binary attributes which are obtained by binary attribute selection method. Briefly, in relative attribute selection method, we convert the binary class-specific and the most related attributes to the relative attributes. During converting process, we use basic rank aggregation methods such as Borda's and Kemeny-Young method to capture the relativity.

We summarize our relative attribute selection method in Figure-4.4. In this method, we first assume that if one or more classes contain an attribute, namely, the labels of this attribute are 1 for these classes, we assign high degree to these classes compared to the remaining classes which do not contain this attribute. Afterwards, we divided all classes into two different groups according to whether the classes contain an attribute or not for that attribute. We process these groups individually. For each group, we generate a set of different ranking orders. If there are k different classes, we can generate $k!$ different ranking orders. And than, we randomly select predetermined number of ranking orders from this set. Furthermore, we aggregate these ranking orders by using rank aggregation methods and we obtain one final ranking order for each group. In addition, if there are less than 3 classes in one group, we randomly assign a number to these classes instead of performing any rank aggregation method. Eventually, final ranking order is obtained by combining two different group's ranking orders.

As seen in Figure-4.4, we capture the relation of equality, greatness and smallness among classes owing to our method. Our generated relative attributes are similar to the attributes which are used in [Parikh, 2011]**.**

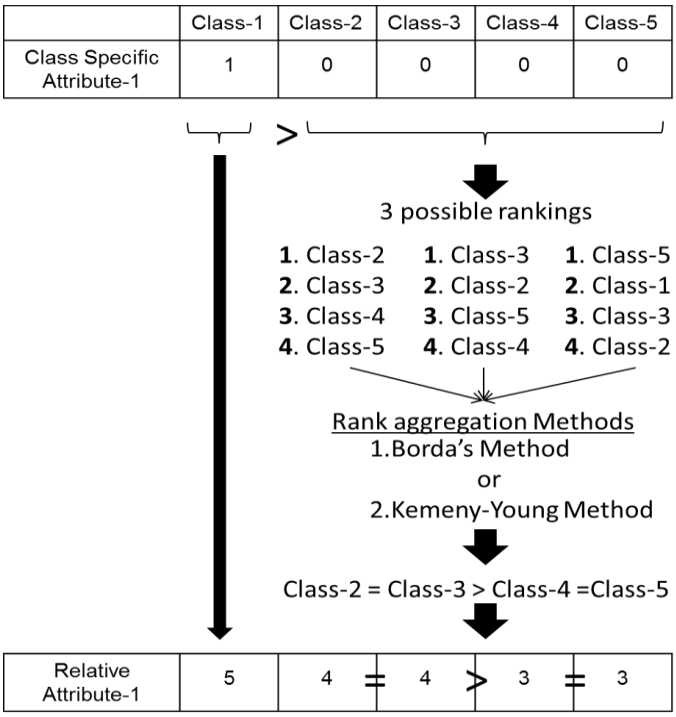


Figure-4.4. The flow chart of obtaining relative attributes from binary attiributes.

During the implementation of relative attribute selection method, we select 30 different ranking orders randomly and we combine them by using rank aggregation methods.

CHAPTER 5

FEATURE EXTRACTION AND CLASSIFICATION

In this chapter, we briefly explain the feature extraction and classification methods used in this thesis.

In our implementation, we use the same features provided by [Parikh, 2011]. These features are gist descriptor and Lab Color Histogram. We explain gist descriptor and Lab Color Histogram in section 5.1 and 5.2 respectively. Followed by, we describe K-Nearest Neighborhood classifier for K-shot classification in section 5.2 and Support Vector Machine classifier for multi-class classification in section 5.3.

5.1 Gist Descriptor

The “gist” is an abstract representation of the scene that spontaneously activates memory representations of scene [Friedman, 1979]. Gist descriptor is originally created for recognition of similar scenes like mountain, street, tall buildings, streets, etc. in [Olivia, 2001]. The shape of whole scene is extracted like a single object, based on Fourier Transform. It is very efficient to compute and tries to capture the global shape of the image although it contains local information to some extent which derives from segmenting the images with a grid.

In detail, firstly, image is divided into n x n split. Secondly, Foriuer transform is performed over each split on the basis of different scales and

orientations. Finally, all scores which is obtained from each split are constitute the gist descriptor features vector. Overview of Gist descriptor is shown in Figure-5.1. The detailed information about gist descriptor can be found in [Olivia, 2001].

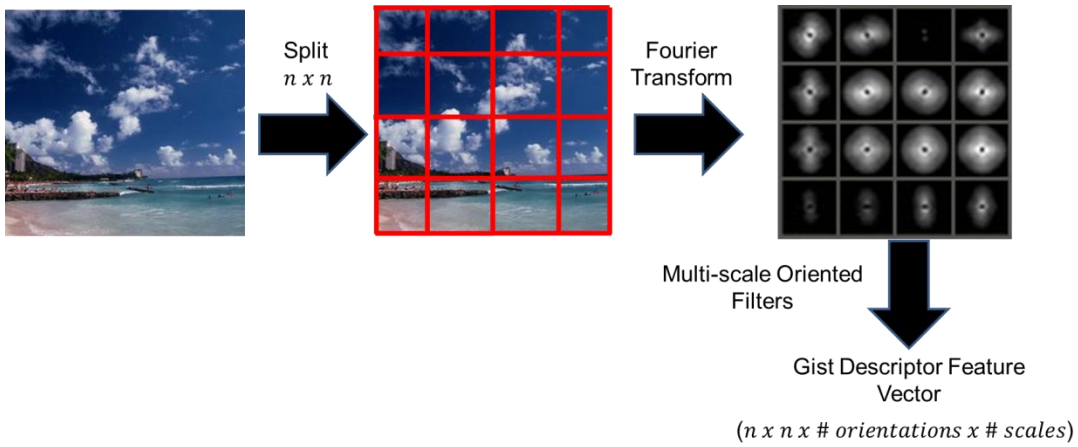


Figure-5.1. Overview of Gist descriptor.

The most important advantage of gist descriptor is that the size of final obtained feature vector is relatively small compared to the other descriptors. In addition, image categories are represented at discriminative level relative to each others with gist descriptor feature vector.

5.2 Lab Color Histogram

Color histogram is a representation of the distribution of colors in an image. Namely, it basically represents the number of pixels that have colors in each of a fixed list of color ranges, that span the image's color space, the set of all possible colors.The color histogram can be built for any kind of color space, although the term is more often used for space like RGB.

Color spaces are the n-dimensional spaces in which the colors can be represented based on their values. LAB and RGB are both 3-dimensional

color spaces. This means both LAB and RGB consist of a 3 value tuple, where each value of the tuple corresponds to an axis of the 3-dimensional space.

Lab color space is an approximately uniform color space that maps equally distinct color differences into approximately equal Euclidean distances in space. In this space, “L” defines lightness, ”A” denotes red/green chrominance and “B” the yellow/blue chrominance.

Presently, it is one of the most popular color spaces for color measurement. Because equal distances in the color space correspond to equal color differences, as perceived by humans, distances between colors in the Lab color space fit better to human perception than distances in RGB color space. In addition, the 'LAB' mode is more useful for determining the actual color balance of an image. The detailed information about Lab color histogram can be found in [Liapis, 2004].

5.3 Classification

At the final step, we use a discriminative classifier SVM for multi-class classification and KNN for K-shot classification to assign a category label which is learnt from the training set to an unobserved image from the test set. Here are the brief details about these classifiers below.

5.3.1. K-Nearest Neighborhood (KNN)

KNN algorithm is the simplest algorithm of all machine learning algorithms. It is an instance-based learning algorithm where density function is only approximated locally. The key idea is to classify a query point in the d-dimensional feature space by a majority vote of its k neighbors where k is

positive integer. Majority voting refers to taking the most common class label among k points from train set which are most similar to the query point.

In details, determining the class of a query point consists of calculating Euclidean distance between a query point and all points in the training set (i.e. class labels are already known for training set), selection of K-nearest points to query point in the training set, and finally assigning the query point to most common class among its K-nearest neighbors. The algorithm needs no explicit training process to create models to reduce the feature dimension. So we need to take all training points into account for computation.

In our experiment, we take the class label of the nearest point of the most representative classes. In addition, we increase the number of instances which belong to unseen in a range (1-30) in testing phase.

5.3.2. Support Vector Machines (SVM)

Support Vector Machine (SVM) is a binary classifier, meaning it can evaluate data points and assign them one of two classes. In order to perform this classification, SVMs are trained with training data and their class labels, and then later assign a class to novel data. Birefly, given a training set of instance-label pairs $(x_i; y_i)$; $i = 1, \dots, l$, where $x_i \in R^n$, $y \in \{1. -1\}^l$, the support vector machines (SVM) [Boser, 1992], [Cortes and Vapnik, 1995] require the solution of the following optimization problem [Hsu, 2010] (C > 0 is the penalty parameter of the error term):

$$\min_{w,b,\varepsilon} \frac{1}{2} w^T w + C \sum_{i=1}^{l} \varsigma_i \tag{5.1}$$

$$s.t. \quad y_i(w^T \phi(x_i) + b) \geq 1 - \varsigma_i \tag{5.2}$$

$$\varsigma_i \geq 0 \tag{5.3}$$

In the training process, the SVM takes a set of d-length vectors (i.e. points of data in a d-dimensional space) and associated classes, while testing requires just a set of vectors. The decision function for a test sample x has the form :

$$g(x)=\sum_i \alpha_i y_i K(x_i, x)-b \tag{5.4}$$

where $K(x_i, x)$ is the response of the kernel function for training point x_i and the test point x, y_i is the class label (i.e. 1 or -1) of the x_i, α_i is the learnt weight of the x_i, and b is the learnt threshold parameter. An illustration of SVM algorithm is displayed at Figure 5.2.

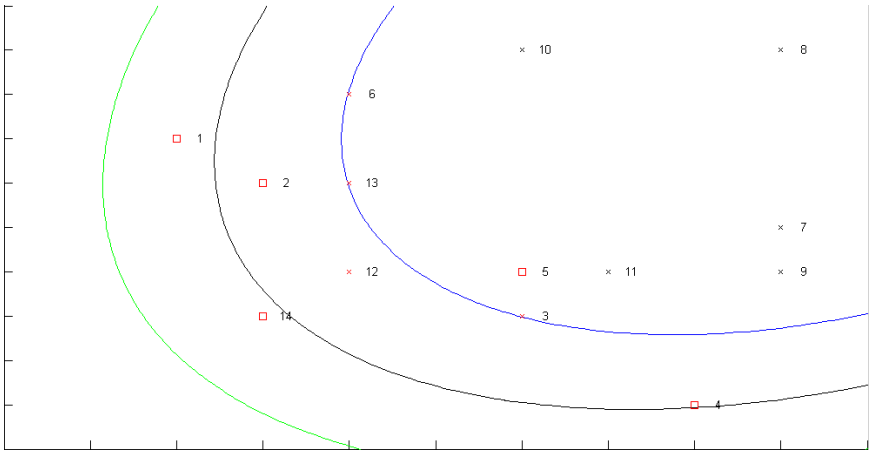


Figure-5.2. Illustration of SVM algorithm.

Training the SVM involves solving a quadratic programming (QP) problem that has as many variables as we have training points. For example we have 13 training points which belong to 2 classes at Figure-5.2, square and cross. At this point, training a SVM over these points means finding the Support Vectors - points that are on the boundary between the classes +1 (i.e. squares) and -1 (i.e. cross). In the Figure-5.2, the actual decision boundary that separates the positive from the negative points is plotted in

black, the contour lines plotted in blue and green are the lines of distance +1 and -1 from the decision boundary which are formed by support vectors (points 3,6 and 13 in this example). All points that are in the margin (between the +1 and -1 lines) are seen as misclassifications.

The choice of a good kernel function $K(x_i, x)$ is very important for SVM learning algorithm. There are many types of general purpose kernels in the literature like linear, polynomial and Radial Basis Function (RBF) kernels. RBF kernel is one of the most popular kernels in the literature which uses a Gaussian distribution to map data to higher dimensions. Namely, RBF kernel nonlinearly maps samples into a higher dimensional space so it, unlike the linear kernel, can handle the case when the relation between class labels and attributes is nonlinear [Hsu, 2010]. Once the data is in this larger space, a hyperplane can be found that separates the data. Generalized form of RBF kernels is [Bennett, 2000]:

$$RBF\ Kernel = e^{-\gamma\, d(x,y)} \tag{5.5}$$

$$d = ||x - y||^2 \tag{5.6}$$

In our classification problem, we have 8 categories. So, instead of solving the binary classification problem, a one-vs-all method is employed to train SVMs for this multiclass problem. In this scheme, each SVM is trained to discriminate one class (+1) from all others (-1); in testing, mixture of topics vectors $P(z \setminus d)$ are run through each SVM, and the classifier category with the strongest response is selected as the winner label for the unobserved test data.

Furthermore, to avoid the overfitting problem, k-fold cross validation is commonly used method. In k-fold cross-validation, training data is first partitioned into k equally (or nearly equally) sized folds. Subsequently, k iterations of training and validation are performed such that within each iteration a different fold of the data is held-out for validation while the remaining $k - 1$ folds are used for learning.

In SVM with RBF kernel, there are two parameters: C and γ that must be specified. In order to specify these parameters, we use “grid search” on C and γ parameters using k-fold cross validation. Grid-search is a commonly used problem-solving technique that consists of systematically enumerating all possible candidates for the solution and checking whether each candidate satisfies the problem's statement. Consequently, since we have only two parameters, we use “grid method” to specify these parameters.

CHAPTER 6

PERFORMANCE EVALUATION

In this chapter, we evaluate our classification methods on two public datasets and compare our results with the most successful methods used in attribute based image/object recognition literature. We first describe the dataset in section 6.1, and then implementation details are given in section 6.2. Finally, we explain our test results and compare ours with current methods which used the same dataset in section 4.3.

6.1 Datasets

We have used two datasets: the Outdoor Scene Recognition (OSR) Dataset [Oliva, 2001] containing 2688 images of 8 categories, and Public Figure Face Database (Pubfig) dataset [Kumar, 2009] containing 772 images from 8 categories. The distribution of images for datasets is shown in Table-6.1. Example images from OSR and Pubfig dataset are displayed at Figure-6.1 and 6.2 respectively. These datasets are used in [Parikh, 2011] and [Ma, 2012] which are the most recent studies in attribute based object recognition literature. Most of the outdoor scenes in OSR dataset display large intra-class variability, meaning that object contents within a scene category are very different. This issue makes the object classification problem harder when working with OSR dataset. Unlike OSR dataset, Pubfig dataset has more stable structure. Size of each image is 256 x 256 pixels.

Table-6.1. The distribution of images for datasets.

OSR Dataset	Coast	Forest	Hihgway	Insidecity	Mountain	Open Country	Street	Tall Building
	360	328	260	308	374	410	292	356
Pubfig Dataset	Alex Rodriguez	Clive Owen	Hugh Laurie	Jared Leto	Miley Cyrus	Scarlett Johansson	Viggo Mortensen	Zac Efron
	95	96	97	94	97	97	98	98

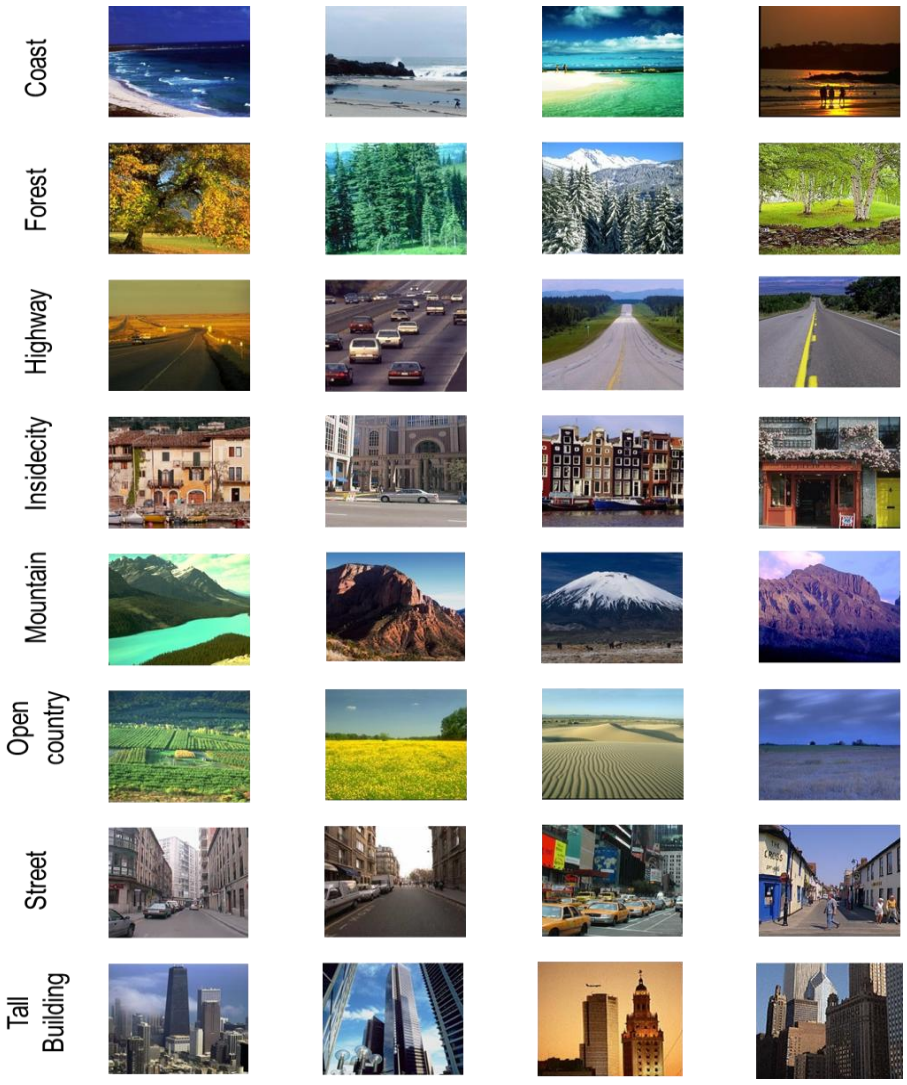


Figure-6.1. Example images from “Outdoor Scene Recognition” dataset.

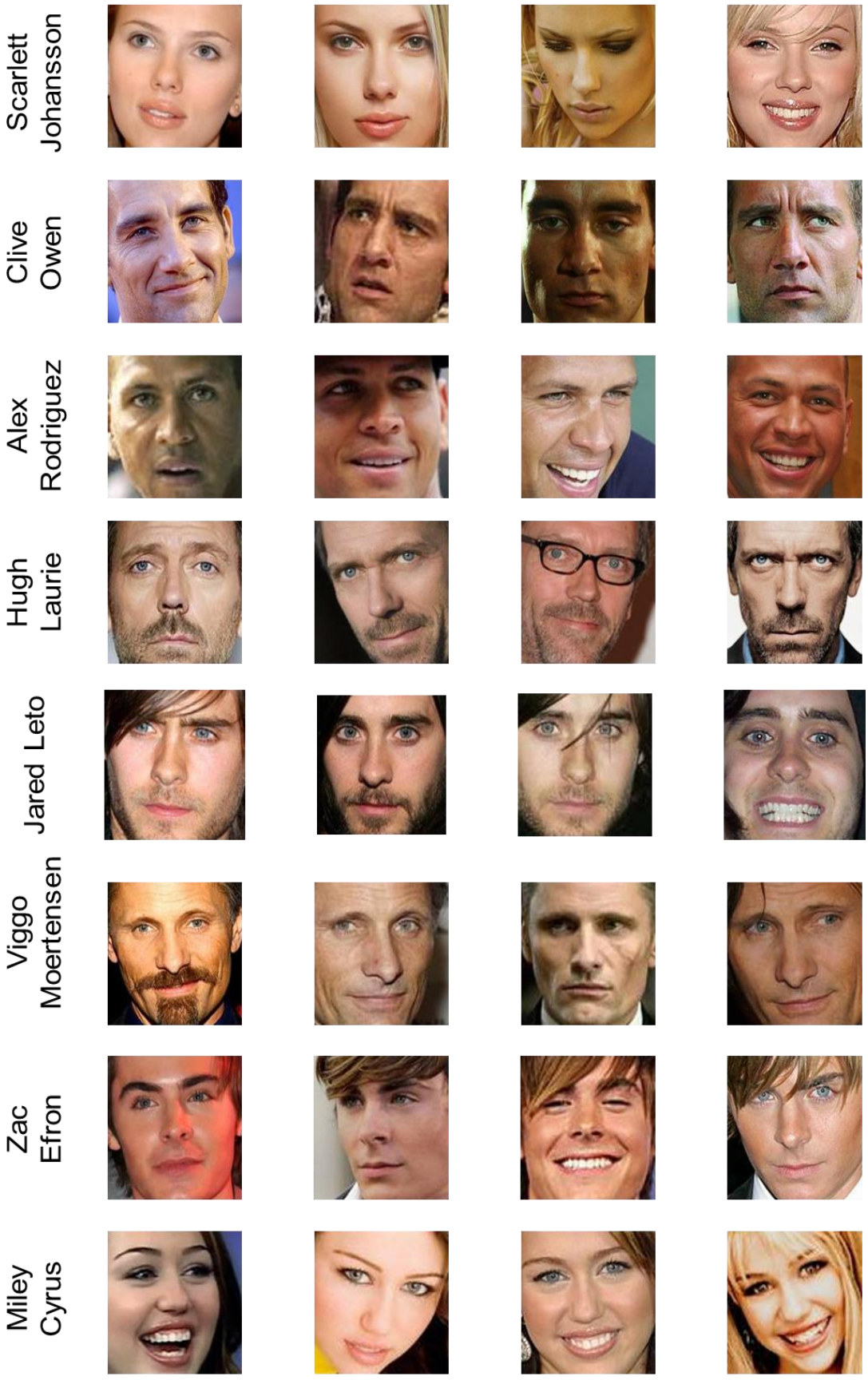


Figure-6.2. Example images from “Public Figure Face” dataset.

6.2 Implementation Details

Since our experimental results are compared with [Parikh, 2011] and [Ma, 2012], we use the same setting which is provided by [Parikh, 2011]. Namely, in oder to perform the comparisons in an equal condition, we use same example images and features with [Parikh, 2011] and [Ma, 2012] for train and test stages.

In our experiment, only 30 example images are used for each image category for training and all remainings are used for test in multi-class classification. In addition, we use different number of training instances for K-shot classes in K-shot classification. We use 512-D gist descriptor for OSR and 542-D feature vector which is combination of 512-D gist descriptor and 30-D LaB color histogram for PUBFIG.

In our experiments, we have used LibSVM [Chang, 2005] with radial basis function (RBF) kernel. Since our data is complex and is not linearly separable, we choose RBF kernel. Furthermore, in order to find C and γ parameters of SVM-RBF kernel, we perform a “grid-search” method on C and γ using 3-fold cross-validation.

Furthermore, we use K-Nearest Neigbour (KNN) classifier for K-shot classification. In K-shot classification, 2 classes (K-shot classes) were left out and attributes were trained on the other 6 classes. We use K (K= 1, 5, 10, 15, 20, 25, 30) training images for each K-shot classes. We perform 10 experiment by randomly selecting K-shot classes and K training images from each randomly selected K-shot class.

In multi-class classification, we perform 30 different experiments for random binary/relative methods and relative attribute selection method and we report the mean. In addition, we combine 30 different ranking orders which are selected randomly in phase of rank aggregation for relative attribute selection.

In random attribute and attribute selection approaches, we use Direct Attribute Prediction (DAP) method [Lampert, 2009] for binary attributes and ranking function [Parikh, 2011] for relative attributes.

In [Parikh, 2011], 6 predetermined attributes for OSR dataset and 11 for Pubfig are used. In [Ma, 2012], 25 attributes for OSR and 28 for Pubfig are used. In our study, since we have no difficulty in increasing of the number of synthetic attributes, we use 28 attributes for both OSR and Pubfig datasets.

We also compare our methods with BIN (Training Linear SVM for every pair of image classes), FLD (Fisher's Linear Discriminant) and PAC (Pricipal Component Analysis) methods. The attributes which are generated by these methods can be categorized as unsupervised binary attributes. In BIN method, we obtain attribute weigths by training linear SVM for each pair of classes separately and we get maximum 28 attributes. In FLD method, Fisher's Linear Discriminant method are used for acquiring attribute weights on the basis of each pair of classes separately as in BIN method. In PCA method, we perform pricipal component analysis for original features of images and principal components are used according to the order of magnitude of eigenvalue (maximum first).

In addition, the system is implemented in MATLAB on Intel Core İ3 Duo 2.13 GHz. with 4.00 GB. RAM.

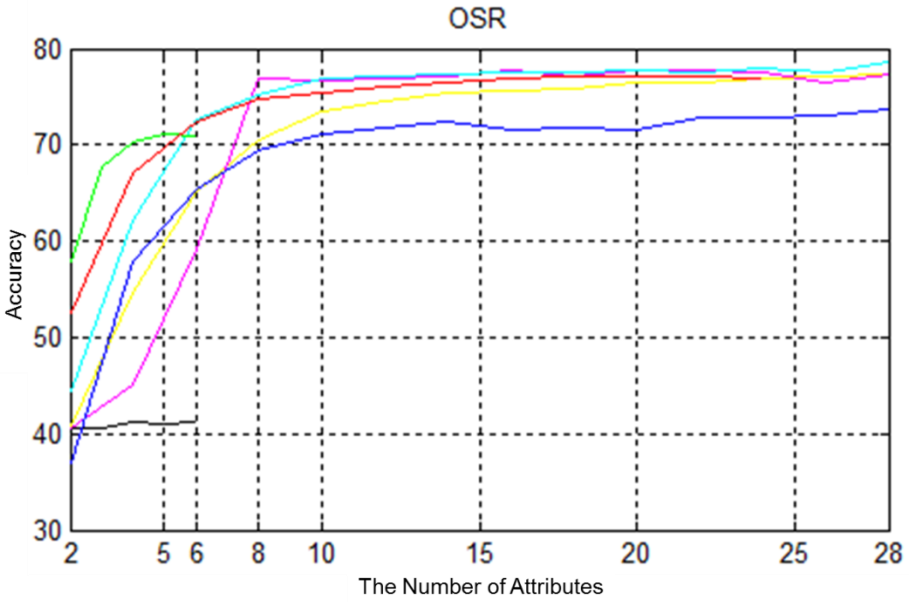
6.3 Experimental Results

As mentioned before, our proposed approaches have been compared to the other attribute based studies in the literature using the same data sets. Methods which are used in Figure-6.3 and 6.4 are; 1) Supervised Binary Attribute Method (DAP) [Lampert, 2009], 2) Supervised Relative Attribute Method [Parikh, 2011], 3) Random Binary Attribute Method (Porposed), 4) Random Relative Attribute Method (Porposed), 5) Binary Attribute Selection Method (Proposed), 6) Relative Attribute Selection Method (Proposed), 7) Unsupervised Relative Attribute Method [Ma, 2012].

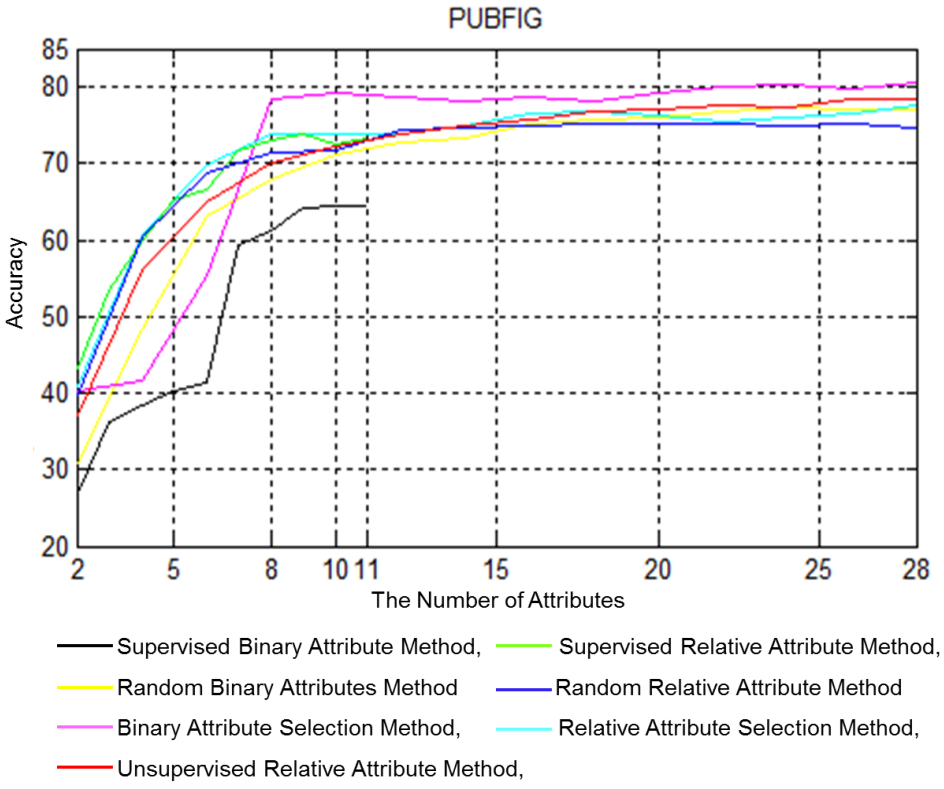
6.3.1 Multi-class Classification Results

Multi-class classification performances are shown in Figure-6.3 for OSR (Figure-3.3.a) and Pubfig (Figure-6.3.b) datasets. In Figure-6.3, it appears that the performance of Supervised Methods are limited by the number of binary and relative attributes: only 6 predetermined attributes for OSR and 11 for Pubfig. In addition, we evaluate that since binary attribute assignments are very similar to each other (detailed in Section-3), Binary Attribute method shows the worst performance compared to other methods.

We also observe that our random binary and relative attribute methods exceed the performance of both supervised binary and relative attribute methods and have nearly the same performance with unsupervised relative attribute method when the number of attributes is increased. We evaluate that performance increase is originated from the increased number of random attributes.



(a)



(b)

Figure-6.3. Multi-class classification results-1.

In addition, attribute selection methods have nearly the best performance compared to the other methods especially at the minimum number of attributes (8 attributes) and they maintain their performance while

the number of attributes is increasing. We evalute that this increase is originated from that attributes are selected more consciously.

Our proposed synthetic attribute methods achieve better performance compared to BIN, FLD and PCA methods as shown in Figure-6.4. We evaluate that since attributes are generated using only two image classes in FLD and BIN methods, it leads poor classification performance.

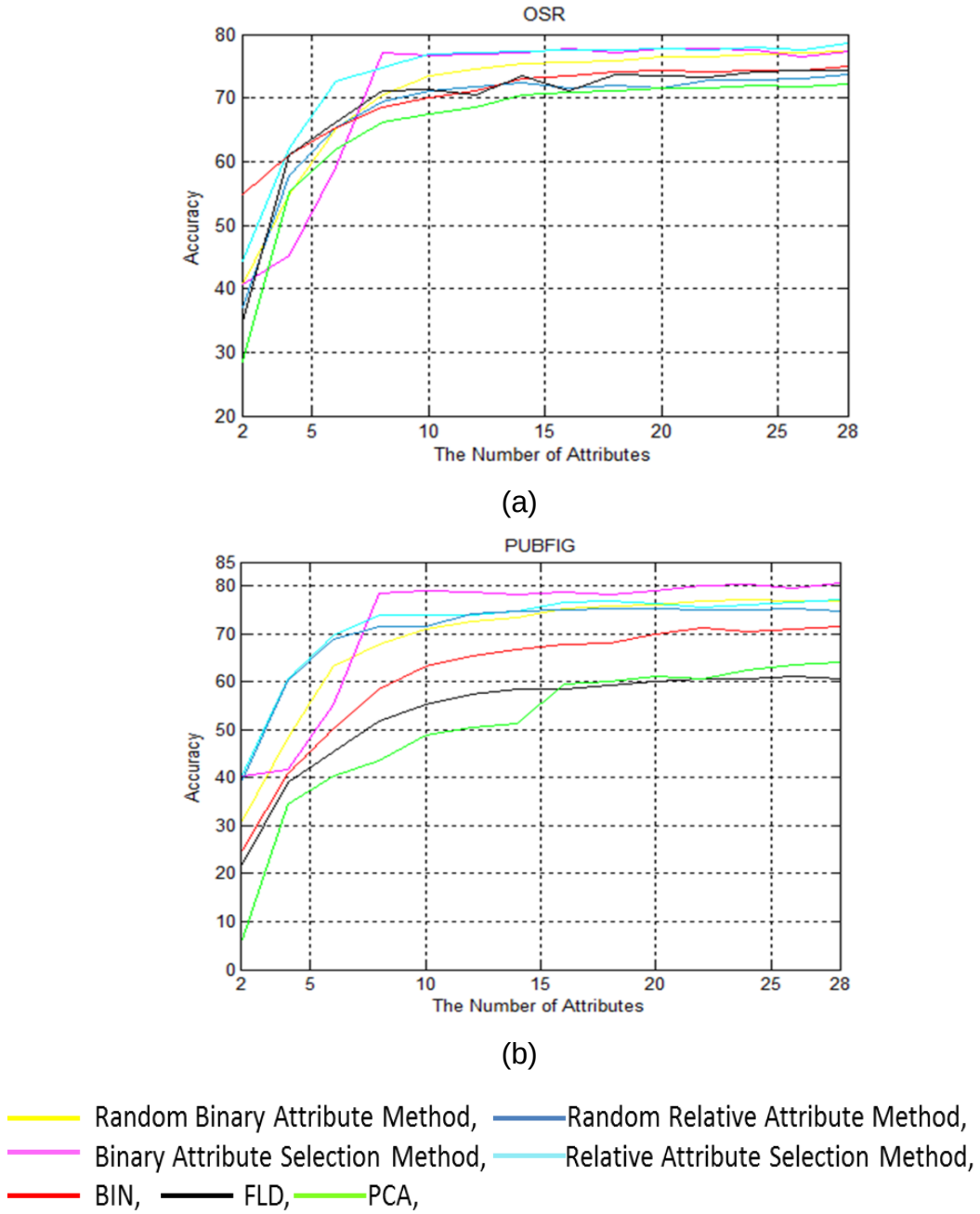


Figure-6.4. Multi-class classification results-2.

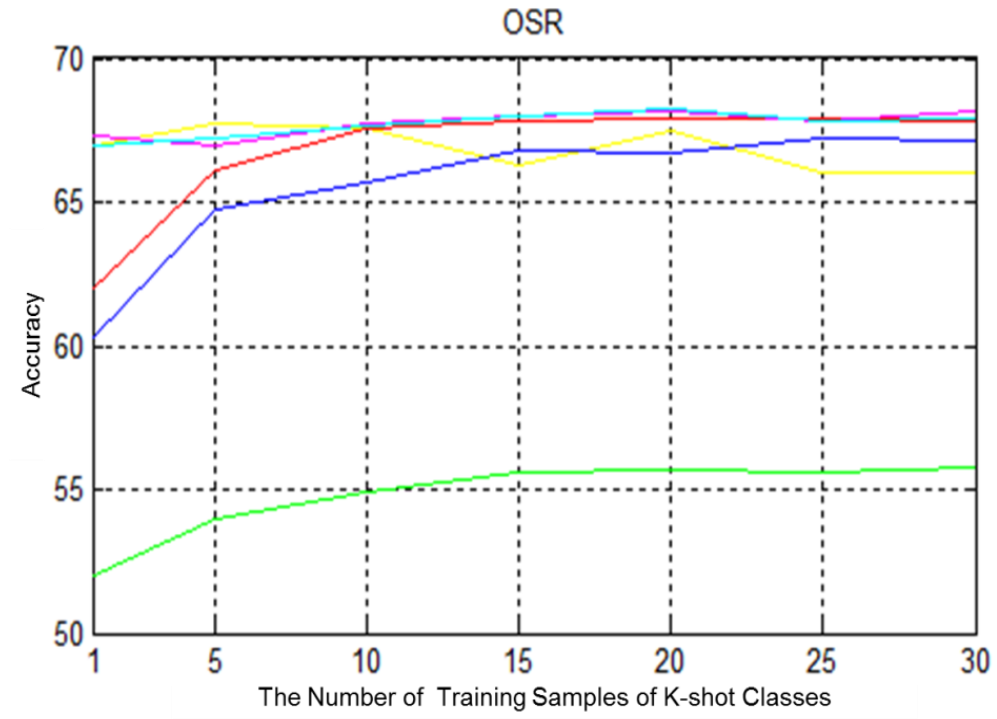
6.3.2 K-Shot Classification Results

We also perform K-shot classification as in [Ma, 2012] for evaluating the generalizability of attributes. We report the mean accuracy in Figure-6.4 for OSR (Figure-6.4.a) and Pubfig (Figure-6.4.b) datasets.

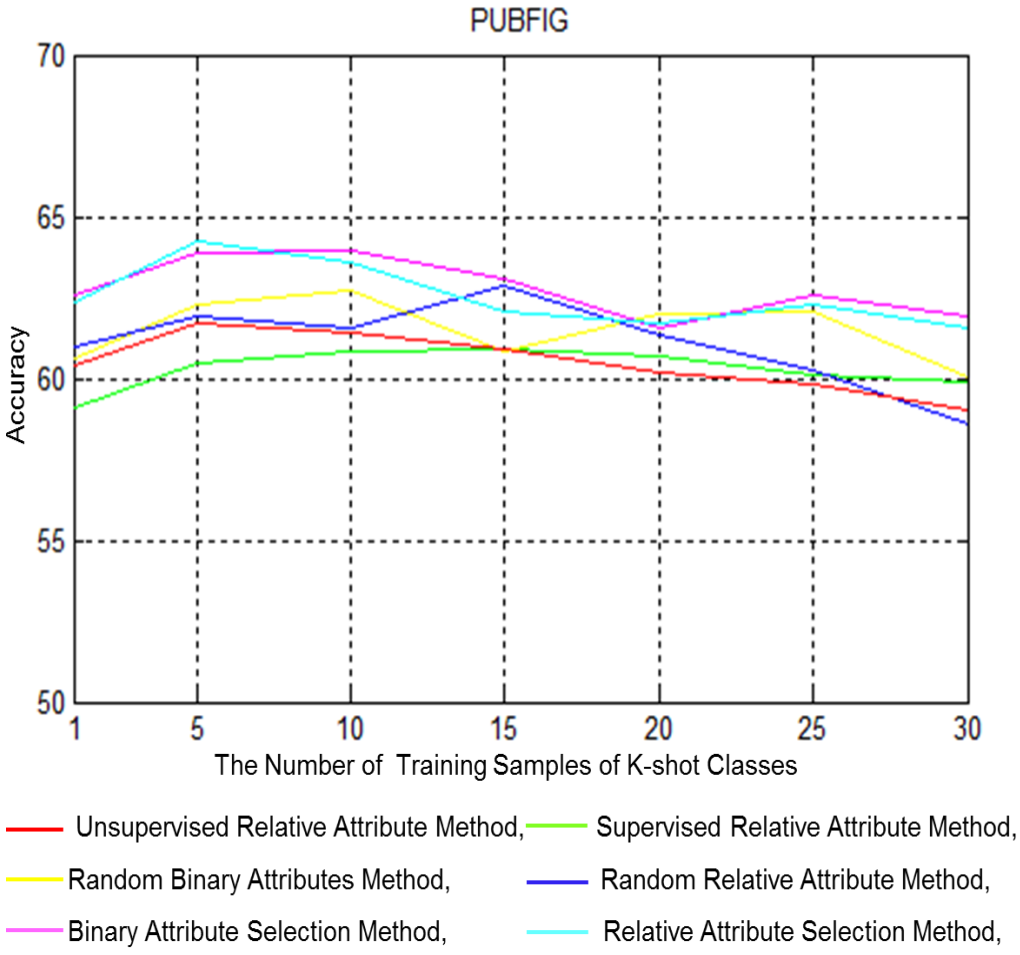
In Figure-6.5, it appears that K-shot classification results confirm the multi-class classification results for both OSR and Pubfig datasets. In addition, Supervised Method shows nearly the worst performances compared to other methods as in multi-class classification. Whether K is small or large, our both random attribute and attribute selection methods show nearly the best performance compared to other methods.

Furthermore, we implement BIN, FLD and PAC methods for K-shot classification, classification results are showed in Figure-6.6. Because attributes are trained using 6 classes in K-shot classification, we generate only 15 $(C\binom{6}{2}; C = combination)$ attributes for BIN, FLD and PAC methods. Similar to multi-class classification results, our proposed synthetic attribute methods achieve better performance compared to BIN, FLD and PCA methods in K-shot classification.

The results indicate that our synthetic attributes have more discriminative capacity for image classes compared to supervised and unsupervised attributes.

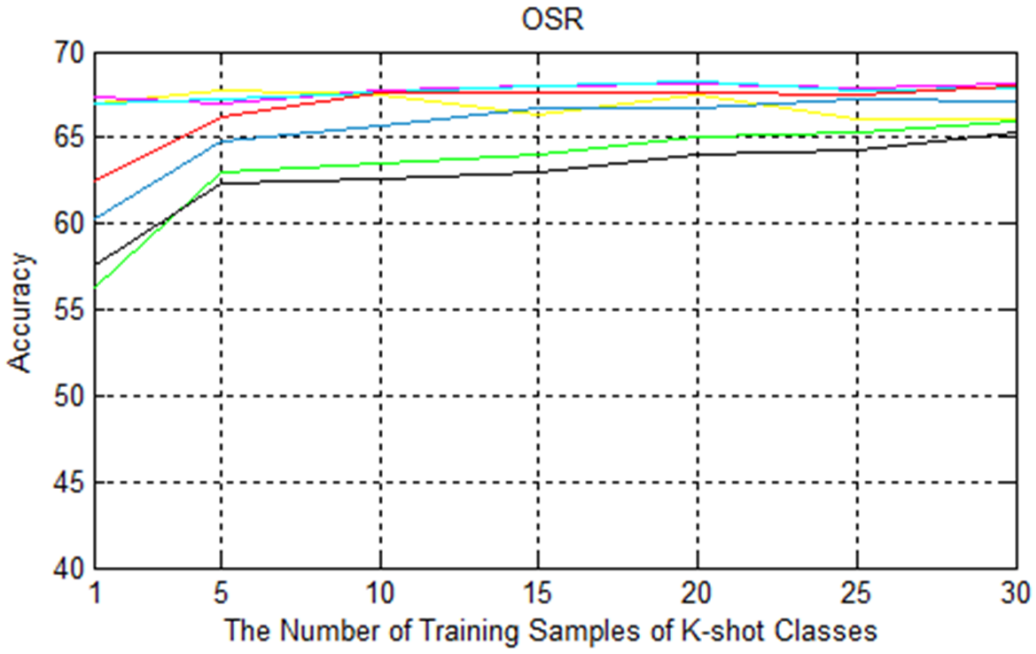


(a)

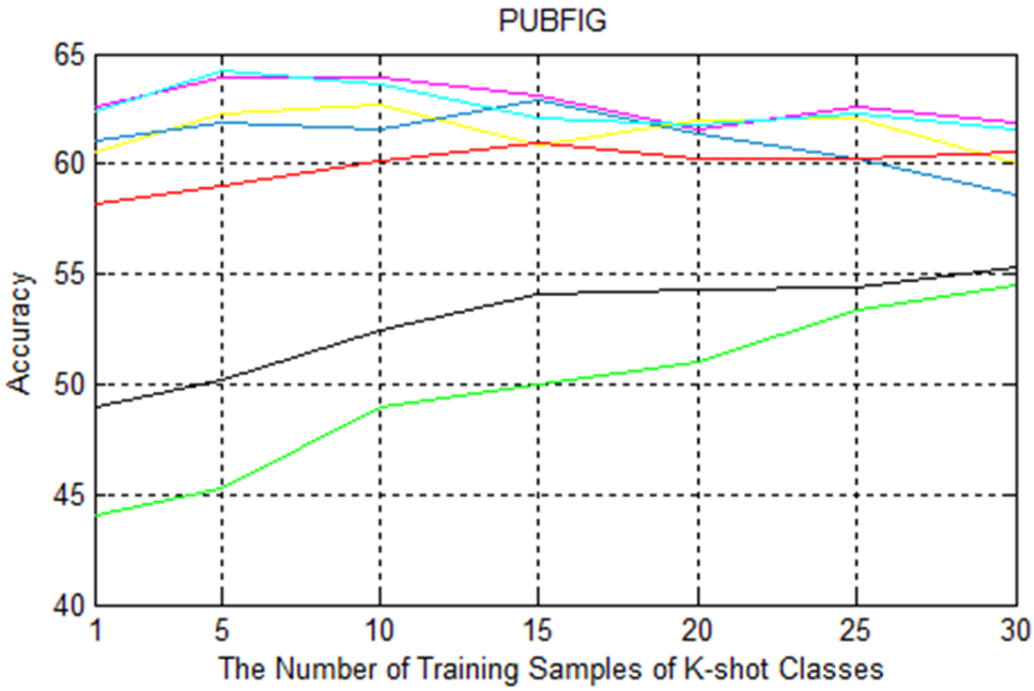


(b)

Figure-6.5. K-shot classification results-1.



(a)



(b)

Figure-6.6. K-shot classification results-2.

Random Binary Attribute Method, Random Relative Attribute Method,
Binary Attribute Selection Method, Relative Attribute Selection Method,
BIN, FLD, PCA,

CHAPTER 7

CONCLUSIONS AND FUTURE WORKS

In general, human being approximate human recognition system more than ever before in the field of computer vision owing to the attribute-based approaches which effectively use the current and widespread machine learning algorithms. In attribute based approaches, objects are described with their semantic properties which are meaningful to human. The notion of attribute transforms the features which do not make any sense to people into meaningful structure.

In addition, human can describe and recognize unknown objects very easily by using their prior knowledge which is obtained according to known objects. But in computer vision, it was very diffucult to classify the unknown objects of which example images are not seen training phase before attibute based approaches. Since many different object classes may share the same semantic properties, attribute based approaches make the process of classification of unknown objects much easier.

In this thesis, we propose synthetic attributes approach to the attribute based object classification. Furthermore, to evalute the generalizability of our attributes and classify the unknown object classes which are not seen in training phase, we perform K-shot classification. In both approaches, we try to solve the problems of supervised and unsupervised attribute approaches and increase the performance of attribute based object classification compare to most current studies in the literature.

Owing to random attributes, we eliminate the exhaustive process of determining of attributes in supervised and semi-supervised approaches. In addition, we do not perform any additional operation in feature space like unsupervised approach. Thus, we accelerate the training process. In addition, we generate the attributes which have more discriminative characteristics relative to each other by random attribute approach. Furthermore, we achieve promising classification performance compared to the other recent studies in attribute based literature by increasing the number random attributes very easily.

Secondly, in attribute selection method, we select the attributes more consciously. The subject of object–attribute interaction is not emphasized in detail in attribute based classification except some studies. We create the relation between object and attributes on the basis of mRMR method in hypothetical space also. The determination of the best attributes in a completely hypothetical space lead to bypass the process of determining of human supervised attributes like random attributes. We also increase the number of most related hypothetical attributes very easily like random attribute approaches. Owing to attribute selection method, we achieve the highest classification performance compared to our random attribute and recent studies. We evaluate that this success is originated from two subjects. First one is selecting the discriminative attributes which depicts the image classes (class-specific attributes). And the second one is the dependency model (mRMR) which is used to find the most relevant attributes with object classes.

Consequently, owing to our proposed approaches, we do not charge any additional load to the system and do not perform any additional operations

in feature space unlike unsupervised relative attribute approach or we do not use any extra datasets or attributes as distinct from unsupervised binary attribute approaches. Since we solve the attribute scalibility problem, we desribe and classify object classes with desired number of attributes.

In context of future works; we will try to generate unsupervised useful attributes by using Deep Learning methods which is more similar to human learning than the current algorithms. We have performed some experiment with Deep Learning but we faced with low classification performance. In addition, so many training examples are needed in Deep Learning methods compare to the current methods. Our aim is to obtain useful attributes while reducing the number of training samples and increase the attribute based classification performance.

REFERENCES

[Aksoy, 2008] Aksoy, S., Boyacı H. and Gökçay D., “The importance of context and semantic descriptions in object recognition: Studies in computer vision and human vision.”, IEEE Signal Processing and Communication Applications, SIU, 2008.

[Andrew, 2004] Andrew Y. N., “Feature selection, L1 vs. L2 regularization, and rotational invariance.”, ICML, 2004.

[Aslam, 2001] Aslam, A., Montague, J. And Momtague, M., “Models for Metasearch”, ACM SIGIR-01, pages 276-284, 2001.

[Bennett, 2000] Bennett K. and Campbell C. “Support Vector Machines: Hype of Hallelujah?," SIGKDD Explorations, vol. 2, 2000.

[Berg, 2010] Berg, T. L., Berg, A. C., and Shih, J., “Automatic Attribute Discovery and Characterization from Noisy Web Data”, ECCV, 2010.

[Biederman, 1987] Biederman, I., “Recognition by components-a theory of human understanding”, Psychological Review, 94(2), 1987.

[Boser, 1992] Boser. B.E., Guyon. I. and V. Vapnik., “A training algorithm for optimal margin classifers.” In Proceedings of the Fifth Annual Workshop on Computational Learning Theory, pages 144-152. ACM Press, 1992.

[Borda, 1781] Borda, J.C., “Memorie sur les elections au scrutin.”, Histoire de L’Academie Royal Des Sciences, 1871.

[Cover, 1991] Cover, T. and Thomas, J., “Elements of Information Theory.”, New York: Wiley, 1991.

[Chang, 2011] Chang, C. C., and Lin, C. J., “LIBSVM : A Library for Support Vector Machines.”, ACM Transactions on Intelligent Systems and Technology”, 2:27:1-27:27, 2011.

[Chow, 1968] Chow, C.K. and Liu, C.N., “Approximating discrete probability distributions with dependence trees.”, IEEE Trans. on Information Theory 14(3), 462–467, 1968.

[Condorcet, 1785] Condorcet , M. J., Essai sur l'application de l'analyse a la probabilite des decisions rendues a la pluralite des voix, 1785.

[Cortes, 1992] Cortes, C. and Vapnik, V., “Support-vector network. Machine Learning”, 20:pages 273-297, 1995.

[Cover, 1974] Cover, T.M., “The Best Two Independent Measurements Are Not the Two Best,” IEEE Trans. Systems, Man, and Cybernetics, Vol. 4, pages 116-117, 1974.

[Dağlar, 2011]Dağlar, M., Güneş, Ö., and Arıca, N., “Probabilistic and Ternary Representation of Attributes in Attribute Based Object Classification.”, IEEE Signal Processing and Communication Applications, SIU, 2011.

[Deng, 2009] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fe, L.: ImageNet: “A Large-Scale Hierarchical Image Database.”, CVPR, 2009

[Ding, 2003] Ding, C. and Peng, H.C., “Minimum Redundancy Feature Selection from Microarray Gene Expression Data,” Proc. Second IEEE Computational Systems Bioinformatics Conf., pages 523-528, August 2003.

[Dwork, 2001] Dwork C., Kumar, R., Noar, M. And Sivakumar D.,”Rank aggregation methods for Web”, 10 th İnternational Conference on the Worl Wide Web, pages 613-622, 2001.

[Ergül, 2009] Ergül, E., “Scene Classification using Spatial Pyramid of Latent Topics”, Master Thesis for Computer Vision, 2009.

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

[Farhadi, 2007] Farhadi, A., Forsyth, D., A., and White, R., “Transfer learning in sign language.”, CVPR, 2007.

[Farhadi, 2010] Farhadi, A., Endres, I., and Hoiem., D., “Attribute-centric Recognition for Cross Category Generalization.”, CVPR, 2010.

[Fellbaum, 2010] Fellbaum, C.: WordNet: “An Electronic Lexical Database.” Bradford Books, 1998.

[Felzenswalb, 2008] Felzenswalb, P., McAllester, D. and Ramanan, D., “ A discriminatively Trained, Multiscale, Deformable Part Models”, CVPR, 2008.

[Ferrari, 2007] Ferrari, V. and Zisserman, A., “Learning Visual Attributes”, NIPS, 2007.

[Fisher, 1936] Fisher, A., “The use multiple measurements in taxonomic problems.”, Annals of Eugenics, Volume 7, Issue 2, pages 179-188, 1936.

[Friedman, 1979] Friedman, A., “Framing pictures: The role of knowledge inautomatized encoding and memory for gist.”, *Journal of Experimental Psychology: General*, 108:316–355., 1979.

[Galleguillos, 2008] Galleguillos, C., Babenko, B., Rabinovich, A. and Belongie, S., “Weakly supervised object recognition and localization with Stable Segmentation”, ECCV, 2008.

[Guneş, 2010] Guneş, O., “Attribute Based Object Classification”, Master Thesis for Computer Vision, 2010.

[Hsu, 2010] Hsu, C.C., and Chang, C.C. and Lin, C.J., “A Practical guide to Support Vector Classification”, National Taiwan University, 2010.

[Ihara, 1993] Ihara, S., ”Information theory for continuous systems.”, Chapter-1, pages. 1-35, 1993.

[Jain, 2000] Jain, A.K., Duin, R.P.W. and Mao, J., “Statistical Pattern Recognition: A Review.”, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 22, No. 1, pages 4-37, Janunary 2000.

[Kemeny, 1959] Kemeny, J., “Mathematics without numbers”, Daedalus 88, pp. 577-591, 1959.

[Kohavi, 1997] Kohavi, R. and John, G., “Wrapper for Feature Subset Selection,” Artificial Intelligence, Vol. 97, Nos. 1-2, pages 273-324, 1997.

[Kumar, 2009] Kumar, N., Berg, A. C., Belhumeur, P. N., and Nayar, S. K., “Attribute and Smile Classifiers for Face Verification”, ICCV, 2009.

[Kwak, 2002] Kwak, N. and Choi, C.H., “Input Feature Selection by Mutual Information Based on Parzen Window,” IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 24, No. 12, pages 1667-1671, December, 2002.

[Lampert, 2009] Lampert, C. H., Nickisch, H., and Harmeling, S., “LearningTo Detect Unseen Object Classes by Between-Class Attribute Transfer”, CVPR, 2009.

[Lazebnik, 2006] Lazebnik, Schmid and Ponce, “Beyond Bag of Features: Spatial Pyramid Matching for Recognition Natural Scene Categories”, CVPR, 2006.

[Legg, 2006] Legg, S. and Hutter M., “A Collection of Definitions of Intelligence.”, Proceedings of the 2007 conference on Advances in Artificial General Intelligence: Concepts, Architectures and Algorithms: Proceedings of the AGI Workshop, pp. 17-24,2006.

[Leibe, 2004] Leibe, B., Leonardis, A., and Schiele B., “ Combined Object Categorization and Segmentation with an Implicit Shape of Model”, ECCV, 2004.

[Li, 2000] Li, W. and Yang, Y., “How Many Genes Are Needed for a Discriminant Microarray Data Analysis?” Proc. Critical Assessment of Techniques for Microarray Data Mining Workshop, pages 137-150, Dececember, 2000.

[Liapis, 2004] Liapis, S. and Georgios, Tziritas, “Color and Texture Image Retrieval Using Chromaticity Histograms and Wavelet Frames”, IEEE Transactions on Multimedia, Vol.6, No.5, October 2004.

[Lowe, 2004] Lowe, D., “Distinctive image features from scale-invariant”, IJCV, 2004.

[Martin, 2004] Martin, D.R., Fowlkes C.C., and Malik J., “Learning to Detect Natural Image Boundaries Using Local Brightness, Color, and Texture Cues.” *PAMI*, pages 530–549, 2004.

[Ma, 2012] Ma, S., Sclaroff, S., and İkizler-Cinbis, N., “Unsupervised Learning of Discriminative Relative Visual Attributes”, ECCV, 2012.

[Moosmann, 2008] Moosmann,F., Nowak E., and Jurie F., “Randomized clustering forests for image classification.”, *PAMI*, 2008.

[Mundy, 1992] Mundy, J. and Zissermann, A., “Geometric invariance of computer vision.”, MIT Press, 1992.

[Mundy, 1994] Mundy, J., Huang C., Liu, J.,Hoffman, W., Forsyth, D. Rothwell, C., Zissermann, A., Utcke, S. and Bournez, O., “MORSE: A 3D Object recognition system based on Geometric İnvariance”, ARPA Image Understanding Workshop, 1994.

[Mundy, 2006] Mundy, J., “Object recognition in the geometric era: a retroperspective”, In Ponce, J., Hebert, M., Schmid, C. and Zissermann, A., editors,”Toward category-level object recognition”, pp.3-29. Springer-Verlag, 2006.

[Oliva, 2001] Oliva, A., and Torralba, A., “Modeling the Shape of the Scene: a Holistic Representation of the Spatial Envelope.”, IJCV, 2001.

[Parikh, 2011] Parikh, D. and Grauman, K., ”Relative Attributes”, CVPR, 2011.

[Peng, 1997] Peng, H.C., Gan, Q. and Wei, Y., “Two Optimization Criterions for Neural Networks and Their Applications in Unconstrained Character Recognition,” J. Circuits and Systems, Vol. 2, No. 3, pages 1-6, Chinese, 1997.

[Peng, 2005] Peng, H.C., Long., F. and Ding, C., “Feature Selection Based on Mutual Information: Criteria of Max-Dependency, Max Relevance, and Min-Redundacy”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 27. No.8., 2005.

[Renda, 2003] Renda, M.E. and Straccia, U., “ Web Metasearch: Rank vs. Score Based Rank Aggregation Methods”, ACM Symposium on Applied Computing SAC, pages 841-846, 2003.

[Riesenhuber, 1999] Riesenhuber, M. and Poggio, T., “Hierarchical models of object recognition in cortex”, Nature Neuroscience, 2(11):1019-1025, 1999.

[Rudas, 2008] Rudas I.,J. and Fodor J., “Intelligent Systems.”, ICCCC, p.132-138, 2008.

[Russakovsky, 2010] Russakovsky, O. and Fei-Fei, L., “Attribute Learning in Large-scale Datasets”, ECCV, 2010.

[Russel, 2006] Russel, B.C., Efros, A.A., Sivic, J., Freeman, W.T. and Zissermann, A., “Using Multiple Segmentation to Discover Objects and their Extent in Image Collection”, CVPR, 2006.

[Saari, 2000] Saari, D.G., “The mathematics of voting : Democratic symmetry”, The Economist, March 4, 2000.

[Shannon, 1948] Shannon C. E., "A Mathematical Theory of Communication", The Bell System Technical Journal, Vol. 27, pp. 379–423, 623–656, 1948.

[Turk, 1991] Turk, M. and Pentland, A., “Eigenfaces for recognition”, Journal of Cognitive Neuroscience, 3(1):71-86,1991.

[Wang, 2009] Wang, G. and Forsyth, D.A., “Joint learning of visual attributes, object classes and visual saliency”, ICCV, 2009.

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

[Webb, 1999] Webb, A., “Statistical Pattern Recognition.”, 1999.

[Yasutake, 2012] Yasutake, S., Hatano, K., Takimoto E. and Takeda M., ”Online Rank Aggregation”, Jounal of Machine Learning Researh (JMLR) W&CP 25:539-553, 2012.

[Young, 1974] Young, H.P., “An axiomatization of Borda’s Rule”, J. Economic Theory, 9:43-52, 1974.

[Yu, 2012] Yu, X.F., Ji, R., Tsai, M.H., Ye, G. and Chang S.F., “Weak Attributes for Large-Scale Image Retrieval”, CVPR, 2012.