8-7-2018

Enhancing Automatic Annotation for Optimal Image Retrieval

Mohamed Masoud

Recommended Citation

Masoud, Mohamed, "Enhancing Automatic Annotation for Optimal Image Retrieval." Dissertation, Georgia State University, 2018.
doi: https://doi.org/10.57709/12501664

ENHANCING AUTOMATIC ANNOTATION FOR OPTIMAL IMAGE RETRIEVAL

by

MOHAMED MASOUD

Under the Direction of Saeid Belkasim, PhD

ABSTRACT

Image search and retrieval based on content is very cumbersome task particularly when the image database is large. The accuracy of the retrieval as well as the processing speed are two important measures used for assessing and comparing the effectiveness of various systems.

Text retrieval is more mature and advanced than image content retrieval. In this dissertation, the focus is on converting image content into text tags that can be easily searched using standard search engines where the size and speed issues of the database have been already dealt with.

Therefore, image tagging becomes an essential tool for image retrieval from large image databases. Automation of image tagging has received considerable attention by many researchers in recent years. The optimal goal of image description is to automatically annotate images with tags that semantically represent the image content. The speed and accuracy of Image retrieval from large databases are few of the important domains that can benefit from automatic tagging.

In this work, several state of the art image classification and image tagging techniques are reviewed. We propose a new self-learning multilayered tagging framework that can address the limitations of current approaches and provide mutual accuracy improvement between the recognition layer and the annotation layer. Our results indicate that the proposed framework can improve the overall accuracy of information retrieval in a variety of image databases.

INDEX WORDS: Image annotation, Histograms of Oriented Gradients, Improved fisher kernel, Deep learning, Fusion algorithm, Rejection principle

ENHANCING AUTOMATIC ANNOTATION FOR OPTIMAL IMAGE RETRIEVAL

by

MOHAMED MASOUD

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

in the College of Arts and Sciences

Georgia State University

2018

ENHANCING AUTOMATIC ANNOTATION FOR OPTIMAL IMAGE RETRIEVAL

by

MOHAMED MASOUD

Committee Chair: Saeid Belkasim

Committee: Anu Bourgeois
Iman Chahine
Raj Sunderramen

Electronic Version Approved:

Office of Graduate Studies

College of Arts and Sciences

Georgia State University

August 2018

DEDICATION

To the soul of my father, to my son Mahdy, and to my daughter Noureen.

ACKNOWLEDGEMENTS

First of all, I thank Allah (God) for giving me the passion, patience, and ability to finish my Ph.D.

I would like to express my deepest gratitude to my advisor, Dr. Saeid Belkasim, for his excellent guidance, support, patience, and suggestions throughout my program. His extraordinary commitment, expertise and effectiveness as a mentor paved the way for me to success in my degree. Definitely, I own him my success in completing such a major milestone. Besides my advisor, I would like also to extend my thanks and appreciation to my committee members Dr. Anu Bourgeois, Dr. Iman Chahine and Dr. Raj Sunderraman for their support and constructive comments during my dissertation work.

Also, I would like to thank Dr. Carol Winkler and Dr. Tony Lemieux from the TCV program for their help and guidance. Many thanks go also to all faculty, staff, and administrators of our department for their help and professional attitude during my study. Special thanks to my lab members as well as to the people of the neuroscience department at Emory University for their collaboration.

Also, words cannot express how grateful I am to my family especially to my mother for her unconditional love and support. I am truly grateful to my elder brother Mahmoud, my sister Hanan and my nephews Ahmed and Abdel Rahman for their love, encouragements and stood by me through the good and hard times.

Finally, my heartfelt thanks go to my wife Noha for her patience, tolerance and for everything that she has made on my behalf over the years. I would like also to thank her family.

Thank you for everyone who supported me during my study, and also my sincere apologies that I could not mention all individually.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 INTRODUCTION

The explosive growth of digital imaging devices (e.g. smart phones, digital cameras) in the past two decades along with the presence of convenient mechanisms to share pictures and videos have led to an overwhelmingly expanding supply of visual data. In 2014, Yahoo reported that over 800 billion photos had been uploaded to the web. This number is expected to grow exponentially every year. Retrieval of visual data based on contents is a challenging task [1-2, 9-10]. Tagging images based on their content is a promising mechanism that can objectively find a proper tag for captured images [22-24]. Automatic image tagging based on image content would have huge impact on archiving, filtering, and retrieval of visual data. Subjective manual annotation is impractical due to the huge visual data produced daily. Therefore, automatic image annotation is a possible solution that aims to bridge the gap between the visual and semantic concepts [39-40]. Automatic annotation is the process that utilizes machine learning techniques to identify proper annotation for partially or fully untagged image by mapping its visual content to semantic predefined concepts. One of the main issues associated with the current image retrieval on the internet is the need to increase the efficiency of the image search engines that deploy tags for locating images. Therefore, tags provide an attractive approach for searching web contents based on text queries. However, to retrieve images with no tags or ambiguous tags, low level image features can be used to help in assigning a sematic tag to an image. Semantic tagging is an active research area where tag recommendation techniques have been widely investigated by many researchers leading to considerable improvement in Tag-Based Image Retrieval (TBIR) [41-42].

1.1 Content Based Image Retrieval

In automatic image tagging system, Content Based Image Retrieval (CBIR) represents the first stage in the system where extracting the low level features of the untagged image is used to represent the image. The low level features are directly extracted from the image pixel information. Similar to data mining and text mining, we can use pixel information to identify useful patterns in images. These patterns can be regions, objects or any correlated features in the image.

The concept of CBIR first appeared in 1992 [1] when image color and shape features are used to retrieve similar images from a database. Since then, the technique becomes popular and has contributed significantly to the field of computer vision. A basic block diagram of the typical CBIR is depicted in Figure 1.1, where the query is not by text as the case with most search engines on the web but by image content as in CIBR model where similar images are retrieved from the low level features database. In this dissertation we will explore the deployment of CBIR model for building an efficient automatic tagging system.

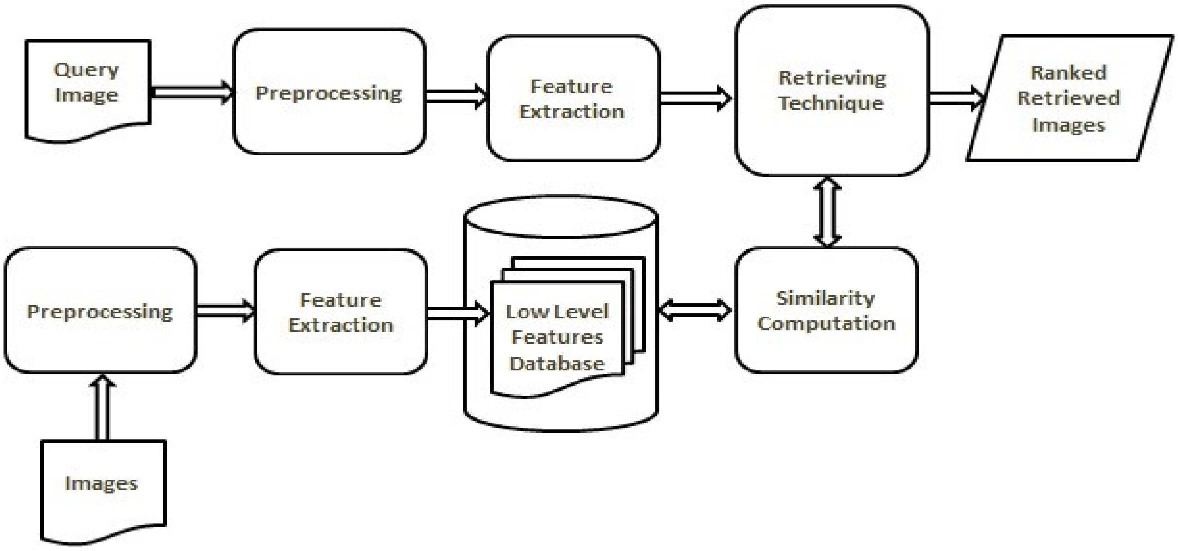


Figure 1.1 Typical CBIR model

The accuracy of CBIR, in general, depends mainly on the choice of the low level feature extraction process. Basically there are three different types of features: color, texture and shape features. Each of these types underline a wide range of methods, and each has its pros and cons. Some of the popular methods are depicted in Fig. 1.2.

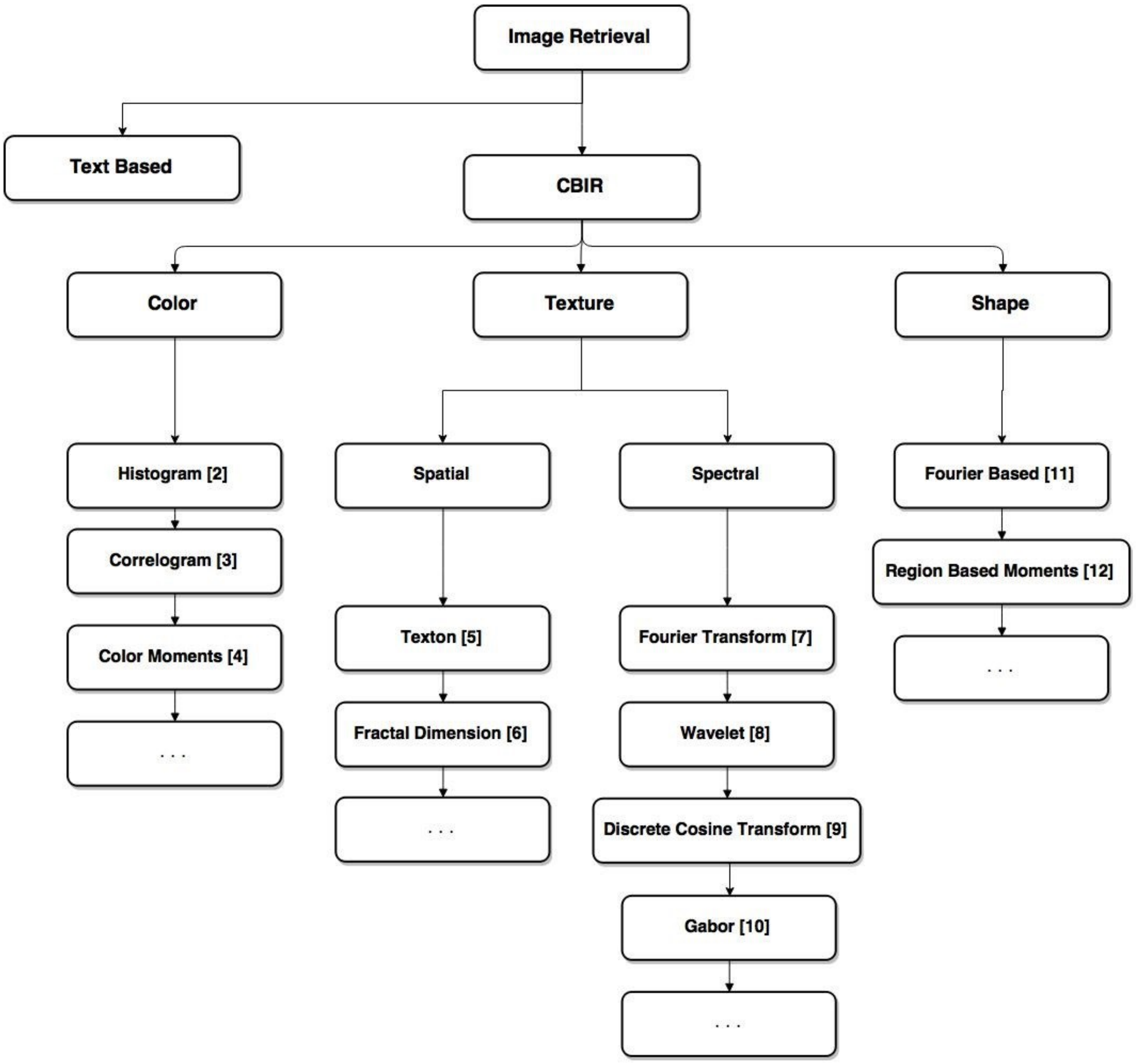


Figure 1.2 CBIR different types and selected methods

In Fig. 1.2, several CBIR techniques are shown. The feature extraction process can be based on one method, or it can be a combination of several methods. Whatever is the feature extraction technique in use, both of the low level features in the database library and the query image must be the same.

1.2 Image Tagging

Image annotation or tagging is used to categorize and indexing images in the form of keywords or labels. In this dissertation, the tagging process addresses the applicability of automatically annotating images with set of keywords that semantically describe each image content. We believe that the improvement in the accuracy of automatic tagging can lead to advancement in many domains such as web image search, image analysis and recognition.

In automatic image tagging, CBIR provides low level features to represent different images. Basically, there are two types of annotations: single labeling, and multi labeling annotation. In multi-labeling annotation, the image is segmented into regions, each region represents a particular scene (e.g. sky, sea, mountains), and/or a particular object (e.g. airplane, ship). There is a need to identify these regions and assign a proper keyword that describes each of them based on a learning model. Although much research has been carried out in automatic image annotation [19 -25], the problem still open and needs further improvement and optimization in terms of accuracy and speed. Also, we are introducing new method to identify semantic relations between generated tags within the same image to predict new tags for other untagged regions and thus increase the overall retrieval accuracy.

1.3 Dissertation Objectives

The main objectives of this dissertation are:

1. To develop a multilayered tagging model that has mutual layer interaction.

2. To use the model generated tags for predicting semantically related tags and improve the overall accuracy of the image retrieval.
3. To build a hierarchical optimization model that can handle multi-learning framework.

This dissertation is organized as follows. Chapter 2 describes Image Tagging and annotation systems. In Chapter 3, we propose our semantic tagging framework. Chapter 4-6 introduce our approaches for single annotation. Chapter 7 describes our multi-label algorithms and results; the conclusions, implications and future research are described in chapter 8.

2 IMAGE TAGGING AND ANNOTATION SYSTEMS

In this chapter, we describe existing image tagging and annotation systems as well as introduce the theoretical background for this dissertation.

2.1 Image Retrieval

Image retrieval techniques are considered to be the foundation of automatic tagging framework, and their performance is directly affected by the efficiency of the low level feature extraction. For example, local descriptors are effective in images that contain many objects, while global descriptors are more effective in natural scene images.

As we mentioned earlier, image retrieval approaches use color, texture, and shape to extract global or local features. The main focus of content based image retrieval for many decades has been on global feature extractors [1] while recently local features gained considerable attention. There is no unified method that can guarantee the best accuracy which makes feature extraction and tagging assignments a challenging problem. Feature extraction based on image colors is able to capture important image information that is invariant to scale and orientation. The simplicity of maintaining image invariance makes color histogram based techniques very attractive for many applications [2].

Texture can be also considered as features for representing certain image characteristics [26]. Fusion of color and texture features has been used in many descriptors as in the micro-structure descriptor (MSD) [27]. MSD can extract low level shape features associated with edge colors, but the retrieval accuracy of this technique showed little improvement.

Shape feature descriptors for binary and gray scale images have been used for image retrieval in techniques such as Fourier-based descriptors [11], and region-based moment descriptors [12]. CBIR is still an active research area that needs more improvement.

In addition to previous techniques, local descriptors show also some advantages such as invariance to image translation, scaling, and rotation. Unlike global descriptors which consider the global image features such as the color histogram, local descriptors breakdown the image into very small regions and consider the properties of these regions (e.g. shape corners) when retrieving similar images. One popular technique widely used is the Scale-Invariant Feature Transform (SIFT) [13].

In SIFT, a computation of the local gradient orientation can be achieved as shown in Fig. 2.1, after that the gradient orientation histograms or the so called SIFT keypoint descriptor can be computed. The Sift descriptor is 128 values for each keypoint.

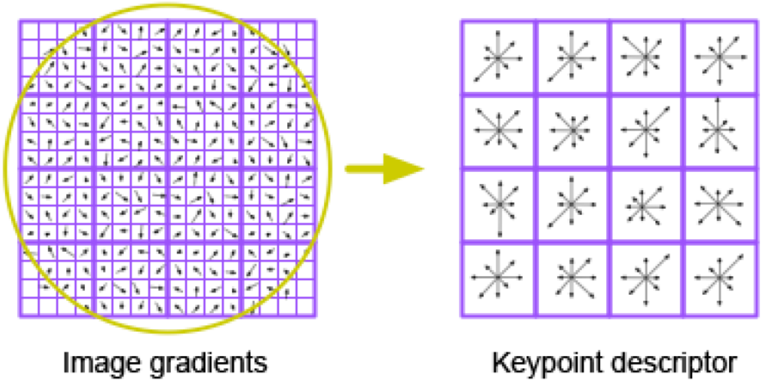


Figure 2.1 SIFT keypoint descriptor [13]

2.2 Image Classification

Image classification is the research area that deals with assigning unknown image to the closest class of a library formed from classified and labeled images. The classification process is dependent on the feature extraction methods that include local or global features. The ultimate task is to achieve class assignment with least errors. Object classification is a special case of the general science of pattern classification and recognition. The pattern can be an object (e.g. person) or an event (e.g. running). Image classification is very sensitive to choice of the feature vector that would be used for the classification process. The feature vectors are the main entries for the learning library which forms the basis for the labeled classes. A distance measure or metric is used to find the closest match in the learning library to classify an unknown image or object. Image regions can be classified based on their extracted features. The library features can be stored as feature vector or optimal set of weights in neural networks. Several techniques are used to represent local and global features within an image among these techniques Bag of Visual Words (BoW), Improved Fisher, and deep learning will be considered in our implementation for their significant performance.

2.2.1 Bag of Visual Words

BoW is a popular local feature based technique for image classification inspired by Bag of Words model that is used in text mining. In document retrieval, the frequency of words in documents are used to classify or retrieve similar documents, the same idea inspired in the image retrieval and classification by using image features as visual words and classify the image based on the histogram of the frequency of visual words. The visual word vocabulary can be generated by clustering the image local features, for example in SIFT descriptors key points are often used as feature vectors. These vectors as we mentioned before are 128 dimensional gradient based feature vector. By finding all SIFT key points in an image and using, for example, K-means clustering, the means of each cluster constitutes a visual word. Therefore, if there are N clusters generated then the dimension of the bag of visual words is N.

In one of the most cited work in this area, Lazebnik et al [28] extended the BoW technique by proposing a method that works by partitioning the image into increasingly fine sub-regions and computing histograms of local features found inside each sub-region as shown in Fig. 2.2.

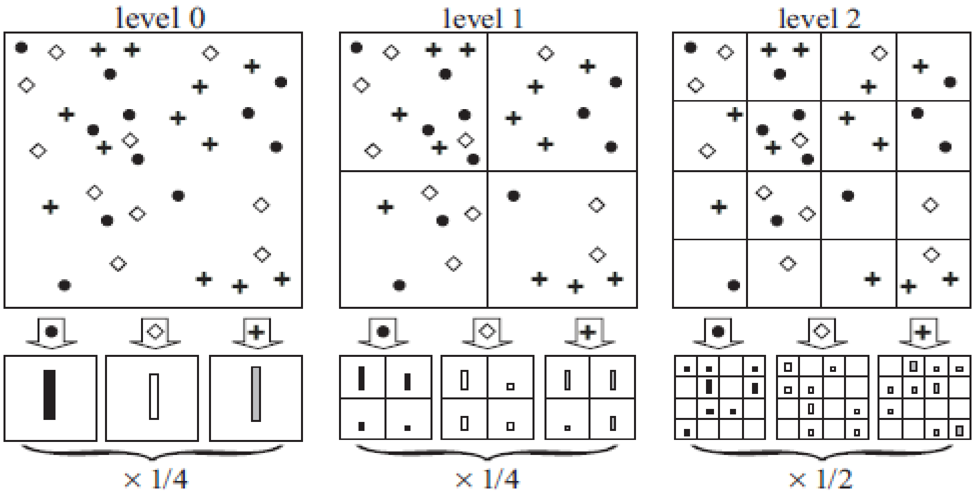


Figure 2.2 Three-level pyramid with different levels of resolution [28]

In Fig. 2.2, a multi-level pyramid is constructed, and in each level the image subdivided into different levels of resolution. This subdivision process resulted in a number of channels; the frequency of features in each channel is counted and represented by spatial histogram.

The pyramid matching mechanism measures the similarity between images in each pyramid level by comparing their channels histogram or the statistics of these histograms such as their means. Therefore, two points in two images are said to be matched if they fall into the same cell of the grid, and the weight of this matching is proportional to the number of levels.

The generated "spatial pyramid" is an efficient extension of the bag-of-features image representation, and it shows more improved performance in scene classification when evaluated on the Caltech-101 database [33]. This database contains from 31 to 800 images per category. Their results show that increasing the number of visual words to 200 along with increasing the number of pyramids level to 2 resulted in increasing the scene classification accuracy based on the support vector machine (SVM) classifier trained using the one-versus-all rule. However, their experimental work shows also there is an optimal number of visual words and pyramid level that can give best accuracy.

2.2.2 Improved Fisher Kernel

As an alternative approach to the BoW technique, Fisher kernel (FK) [34] is a powerful framework that shows competitive performance in the field of image classification. The advantage of this technique is because it combines the strengths of generative and discriminative models. In generative models (e.g. Naïve Bayes, Gaussian mixture model (GMM)) we are given some sample data and labels where the model finds the hidden parameters and specifies the joint probability distribution between the observed and target variables. For discriminative models (e.g. SVM, Neural Networks) which are also called conditional models, they allow only sampling of the target variables conditional on the observed quantities. Therefore, discriminative models focus directly on the classification problem while generative models can handle variable length data.

The main idea of FK approach in image classification domain is to characterize the input images with a gradient vector derived from a generative probability model which is a GMM. The GMM can approximate the distribution of the images low level features such as the visual vocabulary. Therefore, the gradient representation here is replaced the histogram of occurrences that we discussed in BoW section. This replacement results in making the discriminative classifier that received gradient representation can be linear classifier with high accuracy rather than the costly kernel used with the BoW technique. Using the gradient vector can help in determining in which direction the model parameter should be modified to best fit the data. In the context of image classification the FK can actually be understood to extend the popular BoW by going beyond count statistics. A concise comparison between both techniques shows that: Both FK and BoW are based on a visual vocabulary; FK based on GMM while BoW uses K-means clustering, FK properties support the use of linear classifiers while BoW stipulate nonlinear classifiers to give good classification accuracy.

Because of the original FK framework has not shown enough superiority over the BoW, several modifications are run over it to boost the accuracy of the FK in what it called improved FK (IFK) [35]. The improvements added to the IFK are the L2 normalization to the gradient vector, and the Spatial Pyramid approach employed by Lazebnik et al [28] in BoW approaches. The main advantage of the L2 normalization is to remove the dependence of the Fisher vector signature on the image specific information (e.g. image background information). Before this normalization, any two images have the same objects but in different scales will have different

Fisher vector signatures. Therefore, the normalization comes here to solve this issue. In the Spatial Pyramid the classification accuracy increases as it has been shown in BoW section, IFK employs this advantage by replacing the BoW histogram extracted from each grid by the Fisher vector.

A comparison between these methods is shown in table 2.1 on Pascal VOC dataset [43], where an increasing in the Average Precision (AP) from 47.9% in the original FK to 58.3% in IFK is shown.

Table 2.1 Comparison between IFK with state of the arts methods

Method	**AP (in %)**
Standard FK (SIFT)	47.9
Context (SIFT)	59.4
Kernel Codebook	60.5
IFK (SIFT)	58.3
IFK (SIFT+Color)	60.3

2.2.3 Convolution Neural Network

Convolutional Neural Networks (ConvNet) are a special category of artificial neural networks [14-15]. They process data with a trainable hierarchical architectures composed of multiple transformation stages as shown in Fig. 2.3.

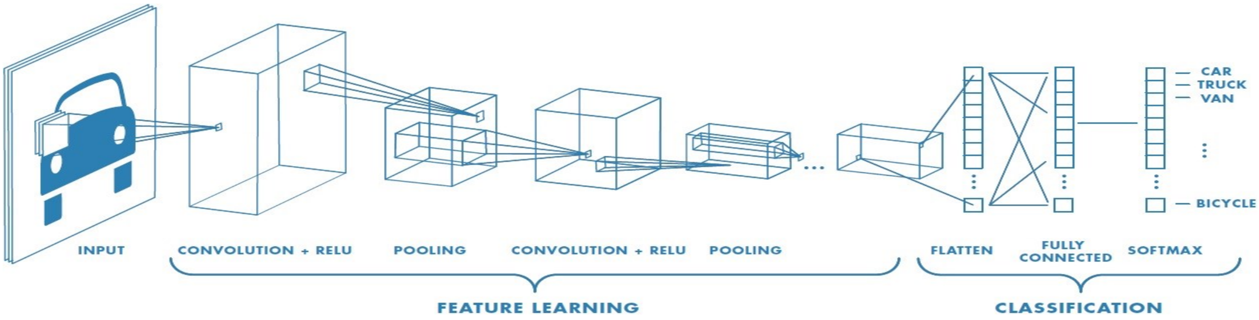


Figure 2.3 Example of CNN architecture (Source: Mathworks.com)

In a typical ConvNet, there are four types of layers that each stage of the network can be composed of: Convolution layer (Conv), Pooling layer (Pool), Fully Connect layer (FC), and Rectifying Linear Unit (ReLU). At the end of the network, there is a classification module to identify the object or image class.

The Conv layer is the layer that does most of the computations in the network. The layer parameters consist of a set of small spatial learnable filters or kernels as shown in Fig. 2.4.

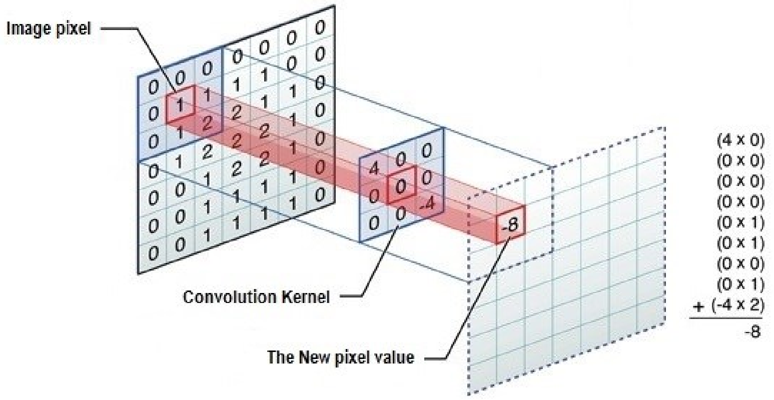
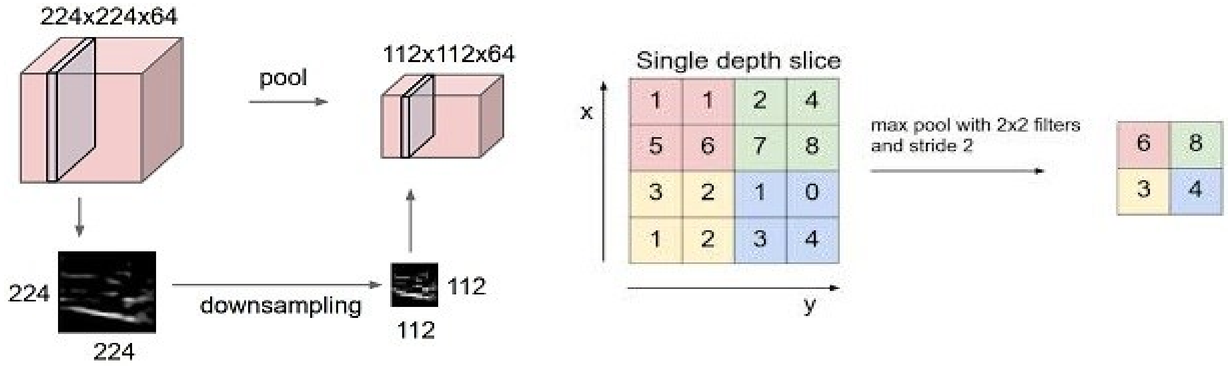


Figure 2.4 Kernel of 3x3 size convolves the input image (Source: developer.apple.com)

Each Conv layer is used to transform a set of feature map to another set of feature map. Therefore the output of each stage represents a particular feature extracted at all locations on the input.

In the pooling layer, a spatial down-sampling of the input feature maps is performed as shown in Fig. 2.5. This down-sampling reduces the number of parameters and computation in the network, and help in making the layer representation invariant to small translation.



1	1	2	4
5	6	7	8
3	2	1	0
1	2	3	4

6	8
3	4

Figure 2.5 The pooling function (Source: cs231n.stanford.edu)

The main advantage of the ReLU layer is to increase the ConvNet training speed by replacing the standard way of modeling the neuron output of tanh with non-saturating nonlinearities function f(x)=max(0 , x) which have been referred to as Rectified Linear Units or ReLU. Using these neuron units the ConvNet can reach 25% training error rate six times faster than standard tanh neurons.

In the FC layer, the neurons have full connections to all activations in the previous layer as shown in Fig. 2.6.

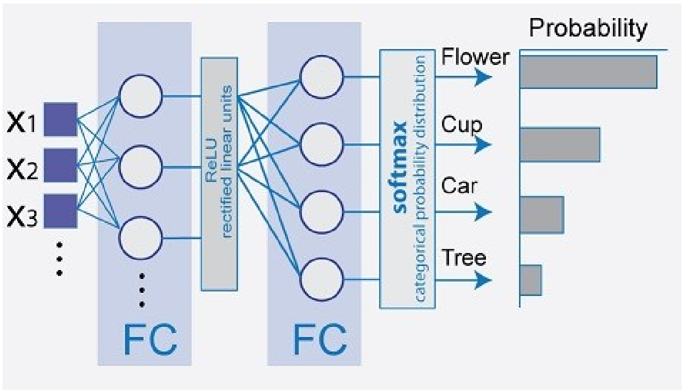


Figure 2.6 ConvNet architecture (Source: Mathworks.com)

The output of this layer can be passed to a softmax function which converts the outputs into class probabilities such that:

$$P(C_k = 1|\boldsymbol{X}) = softmax(f^{out})_k = \frac{e^{f_k^{out}}}{\sum_j e^{f_j^{out}}}$$

where softmax represents the probability of class k given input x, and f_k^{out} is the activation of the k neuron of FC layer.

2.2.4 Semantic Segmentation

Scene parsing, or semantic segmentation, is the process of labeling each pixel in an image with the category of the object it belongs to. It is a challenging task that involves the simultaneous detection, segmentation and recognition of all the objects in the image. In an ideal scene parsing, every region and every object is annotated as shown in Fig. 2.7.

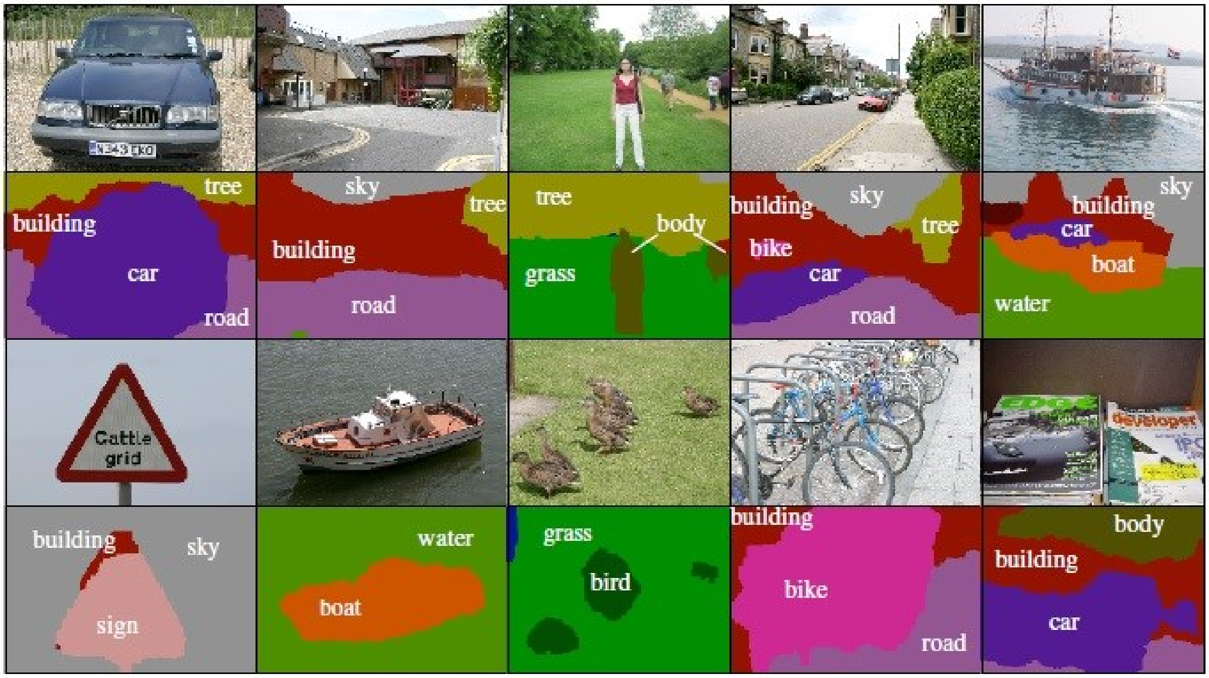


Figure 2.7 Semantic segmentation examples [19]

Scene parsing is associated with many challenges. The difficulty of segmenting and recognizing multi-labels at the same object of the scene, dealing with noise, non-homogenous background, scale, rotation and reflection variations are some of these challenges. The traditional approaches in segmentation are mostly heuristic based. The generated segmentations are encoded and a trained prediction model is used to produce the segmentations label. However, such approaches need sophisticated post-processing to ensure the global consistency between the produced labels.

One technique that recently used successfully in this domain is the ConvNet as in Farabet et al [20]. In ConvNet the raw image pixels are fed into the ConvNet and trained in supervised mode with fully labeled scenes to categorize each pixel location. Using a large contextual window to label pixels reduces the traditional post-processing requirement for consistent labels.

2.3 Performance Evaluation Metrics

In Image retrieval and classification, the performance evaluation can be measured with different metrics. The precision and recall are the most used metrics in these domains. In image retrieval, precision refers to the percentage of true positive (relevant) images in the retrieved images, while the recall is the percentage of true positive images in all relevant images in the dataset such that:

$$Precision = \frac{t_p}{t_p + f_p}$$

$$Recall = \frac{t_p}{t_p + f_n}$$

where t_p is the number of true positive or relevant retrieved images, f_p is the number of false positive or irrelevant retrieved images, and f_n is the number of false negative or relevant non-retrieved images. Therefore, precision measures the retrieval accuracy, whereas recall measures the capability to retrieve relevant items from the database. Another metric that can be used also for the evaluation purpose is the F-score metric which combines precision and recall in different degree according to the score in use such that:

$$F_{score} = (1 + score^2)\frac{Precision\ .\ Recall}{score^2\ .\ Precision +\ \ Recall}$$

Therefore, the increase in the score results in increasing the weight of the recall metric than precision and vice versa. In image classification, the precision of a classifier during the testing phase is the number of true positives compared to the total classified instances.

2.4 Image Annotation

Image annotation techniques can basically be classified into single labelling annotation and multi-labelling annotation. Even though the last one is more dominating, the single labelling can be beneficial if we consider the multi-labelling annotation as a collection of single-labelled regions within the image such that each segmented region can be viewed as an instance and the image as a bag of instances. In this context, we may extend the use of the single-labelling annotation framework shown in Fig. 2.8 to the multi-labelling domain also.

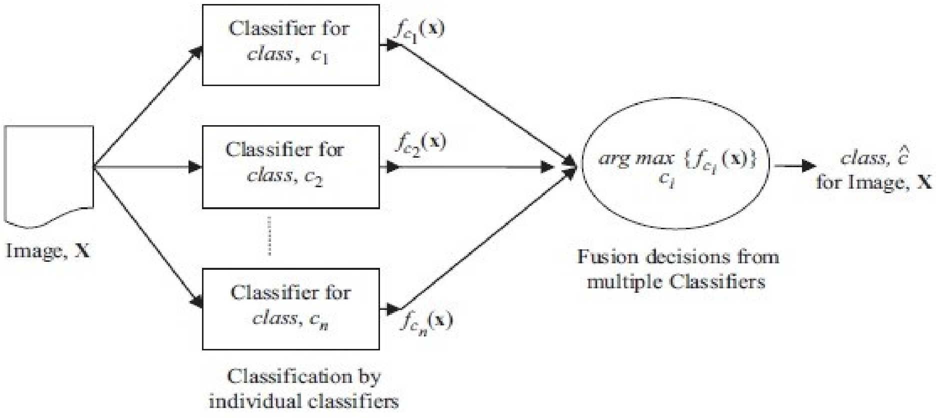

Classifier for class, c_1
$f_{c_1}(\mathbf{x})$
Classifier for class, c_2
$f_{c_2}(\mathbf{x})$
Image, X
Classifier for class, c_n
$f_{c_n}(\mathbf{x})$
$\underset{c_i}{arg\ max}\ \{f_{c_i}(\mathbf{x})\}$
class, $\hat{c}$ for Image, X
Fusion decisions from multiple Classifiers
Classification by individual classifiers

Figure 2.8 Multiclass classifier [36]

Neural Networks (NN) [37] and Support Vector Machine (SVM) [22] as a binary classifier for linear and nonlinear data are among the common classifiers that have been used widely in the single labelling annotation. While the drawbacks of NN are the long time it takes for training and the high probability to fall into local optima, the SVM also has its class imbalance problem in which it performs poor on imbalanced data such as image data.

The multi-labelling approach annotates the image with multiple concepts which is known as multi-instance multi-label (MIML) learning which is different from single labelled annotation. In general, the MIML learning can be based on a probabilistic method (e.g. Bayesian model) where the posterior probability region is assigned to a certain label based on the given observations of specific features extracted from the image region. The model is shown in Fig. 2.9.

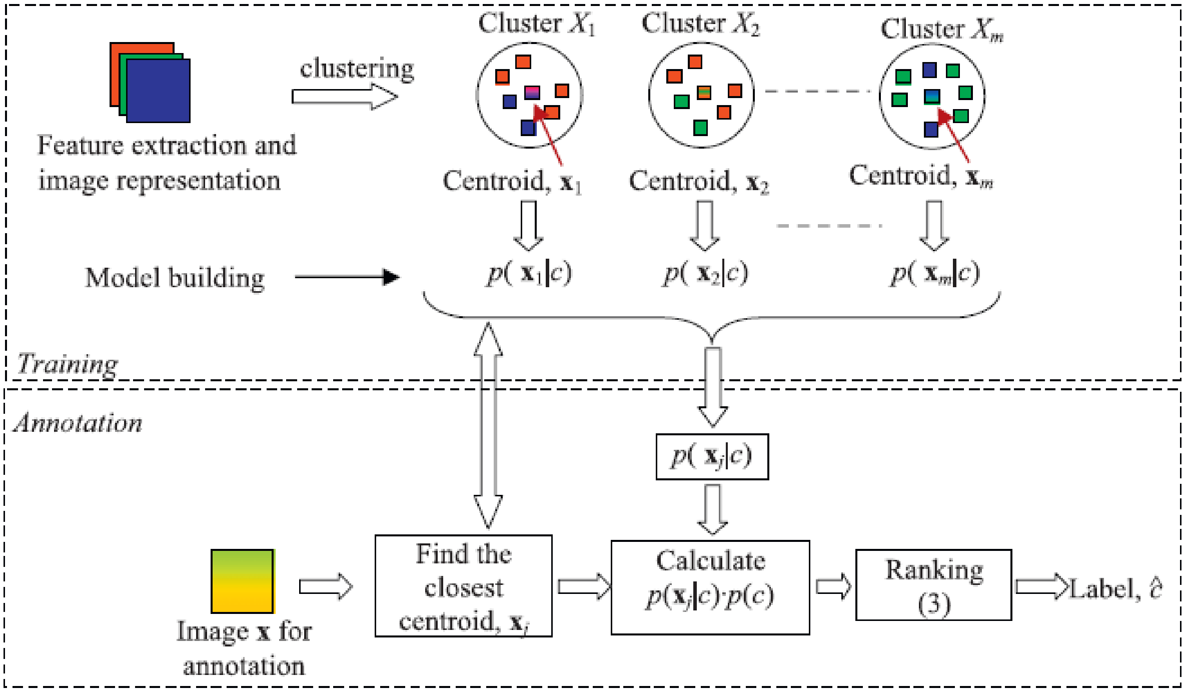


Figure 2.9 Bayesian annotation approach [36]

In Fig. 2.9, x can represent an image or a region feature vector. $p(x|c)$ is the conditional probability of the region feature vector x given the concept c . This conditional probability can be approximated to be $p(x_j|c)$ where x_j is the closet centroid (cluster center). The annotation system can use the conditional probability of the closet centroid to calculate the posterior probability for the region label based on the formula $arg\ max_c \{p(x_j|c)p(c)\}$.

The previous approach is called non-parametric because it does not stipulate any prior assumptions about the distribution of the image or region features. In the parametric approaches, such a distribution is required for computing the conditional probability $p(x|c)$. Carneiro et al [39] use the parametric approach by assuming that the image features follow a certain Gaussian distribution in the feature space, and built a Gaussian Mixture Models (GMM) for each concept by averaging the individual GMMs within each concept as shown in Fig. 2.10.

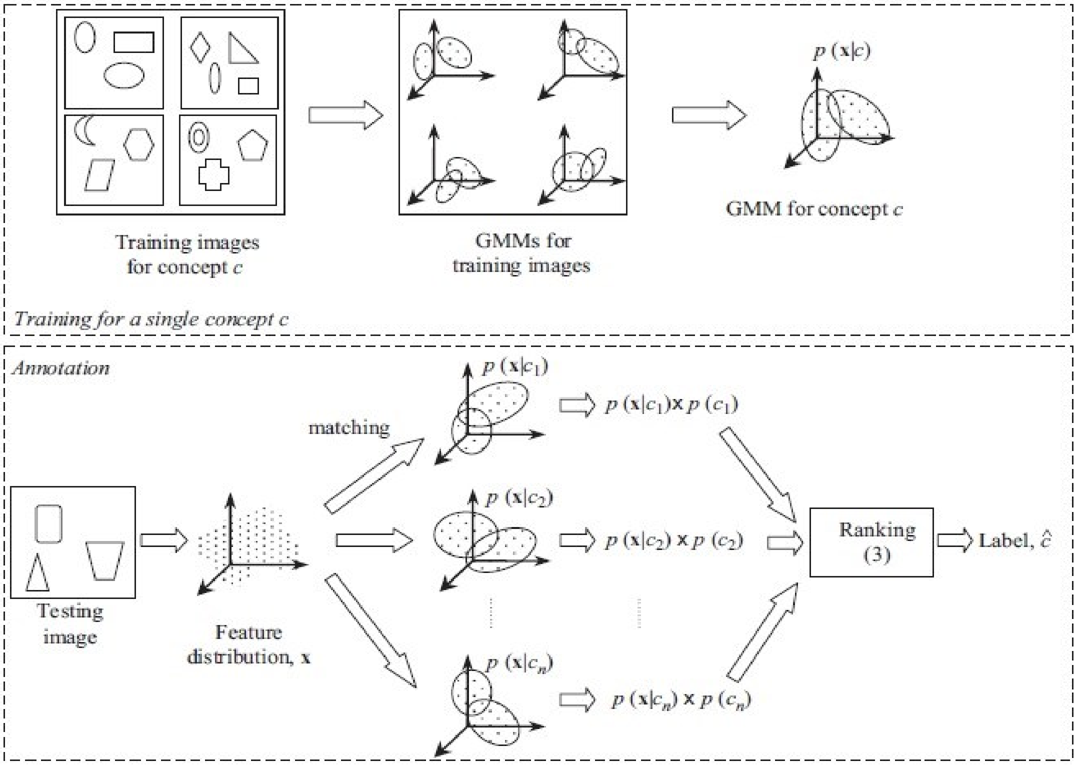


Figure 2.10 Parametric approach using GMM modelling [39]

The drawback of the method in Fig. 2.10 is the complexity in estimating the GMM models.

Many different techniques have been used in annotation systems. Semantic layer tagging [23] is one of these techniques that utilize word embedding techniques for bridging the semantic gap between the image content and the related tag text, where each word is associated with a vector. Vector of pairs, each image and its related text, are used for the model training phase and for optimizing the mapping function in the semantic space where a text component of an image is a linear combination of its tag representation vectors. The mapping function is trained to map the untagged image features to the semantic space to infer its tag. A difference measure is formulated to ensure the consistency between the mapped image features and the tag representation in the semantic space. Other learning-based approaches are also used widely by constructing many discriminative models such as SVM [24-25] to predict image tags from low level features.

Another feature is added in [40] by incorporating keyword correlations and region matching, a heuristic greedy iterative algorithm to estimate the probability of a word being a caption of an image.

However, successful tag assignment requires both efficient retrieval and accurate tagging approaches, and almost all the approaches connected with the annotation domain consider only the modification in the tagging layer without any feedback for adjustment in the retrieval layer.

After producing the image tag, a refining process is needed to make the query more efficient, for example, if we are given an image tag of “apple”, a domain is needed to identify the word sense in this case. In the context of word domain, the word can be apple food or apple computer [41]. Different techniques for tag refining are surveyed in our previous work [42].

2.5 Image Retrieval and Annotation Datasets

In this section, we introduce the most widely used datasets in image retrieval and annotation:

- Pascal VOC 2007 [43] used widely in object and image annotation. It contains 10000 images of 20 classes including animals, handmade and natural objects. Pascal VOC images come with bounding box annotation.

- NUS-WIDE [18] is a popular social image dataset that includes 269,648 images and 5,018 unique tags for images description. The dataset concept taxonomy consists of 81 concepts and several low level features are provided for the evaluation purpose.

- ImageNet [16] is a dataset of around 15 million labeled images categorized in 22000 different classes. A widely used subset of this dataset is the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) which consists of 1000 categories each has around 1000 images.

- Corel 5k [44] is a popular image dataset for image retrieval and annotation. Corel 5k dataset has 5000 images each annotated with1 to 5 words.

2.6 Summary

Most of the current research techniques closely connected to image retrieval, classification, and annotation have been reviewed. However, each technique has advantages and some limitations. SIFT is widely used as low level descriptor with many of the research work done in the annotation area on some large scale databases such as NUS-WIDE, however SIFT is a local shape descriptor and cannot represent the images that can be best represented by global shape descriptor such as nature images as it has been shown in our preliminary work.

In bag of visual words technique the classification relies on the discriminative power of visual vocabulary, it results in the technique limitation of correlating visual words with tags. The same also can be said for the IFV.

The main drawback of ConvNet comes from the high computational power it needs in the training phase which makes this phase impractical without parallel processing. The computational overhead is due to the huge number of parameters it has. This impracticality results in many difficulties in implementing this technique in portable devices (e.g. digital cameras, mobile phones) which are the main sources of captured images on the web.

Almost all current annotation techniques consider the improvement in the tagging phase only without any additional effort for improving the underlying layers. The architecture of these layers and the relations between them are studied in this work and additional improvements are proposed to advance the interaction between the model main components.

3 PROPOSED APPROACH

The number of image retrieval, classification, and annotation techniques has been rapidly increasing over the last decade; however, each technique in any of these categories has advantages and limitations. In chapter two we concisely outlined the main pros and cons of the current popular techniques. The main drawback that is shared with these techniques is that none of them has treated annotation as integrated part of both the classification and retrieval. Most of the research work in annotation area did not consider using some important benefits from classification and retrieval. The classification and retrieval can be used to refine the overall results and give a semantic definition to the annotated category.

In this dissertation we propose to enhance the process of the automatic annotation by utilizing a feedback loop between the annotation of various stages in order to improve the system accuracy and reliability. In this chapter a new multi-layered feedback model is proposed for the annotation process and a description for the overall system architecture is given.

3.1 Multi-level Annotation Framework

In order to design an automatic image tagging algorithm, we propose multilevel framework for the tagging process as shown in Fig.3.1. The proposed framework consists of three main layers: the image retrieval, image recognition, and image tagging.

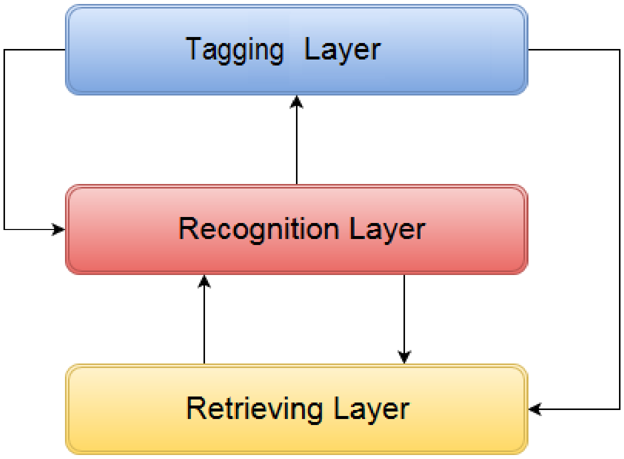


Figure 3.1 Proposed annotation framework

The retrieval layer is used as an input to the proposed system, the middle layer represents the image recognition layer, and the top layer is the image tagging layer. In the recognition layer, the proposed system recognizes the various image regions prior to feeding the output to the tagging layer which assigns the proper annotation. The training block diagram of the proposed model is shown in Fig. 3.2.

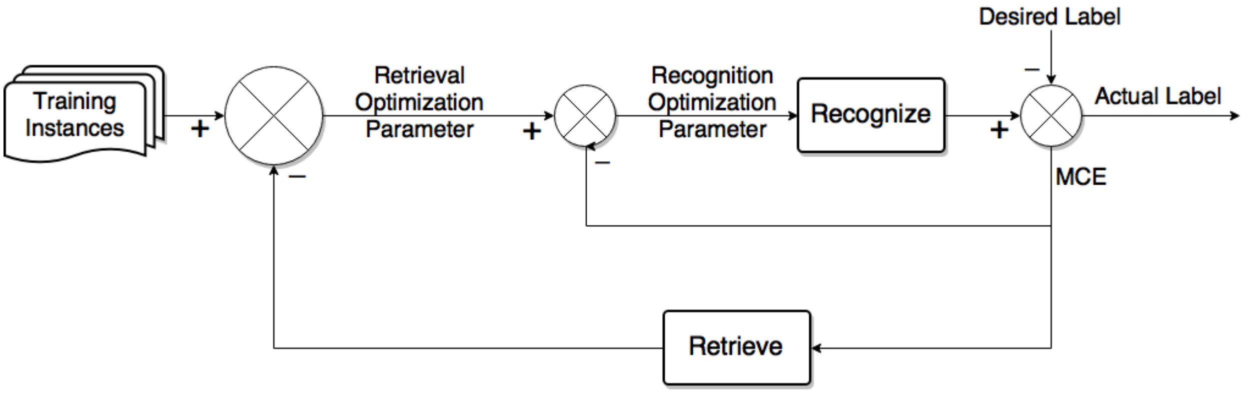


Figure 3.2 Framework training and optimizing block diagram

In Fig. 3.2, we show the basic idea of how to train the annotation framework and optimize the main parameters of the retrieval layer and recognition layer. In the retrieval layer, the optimal parameter depends on the descriptor in use. For example, in the color histogram, the number of bins is the optimal factor.

We propose our multi-label automatic annotation model that implements the feedback between the annotation layers to improve the annotation accuracy as shown in Fig. 3.3.

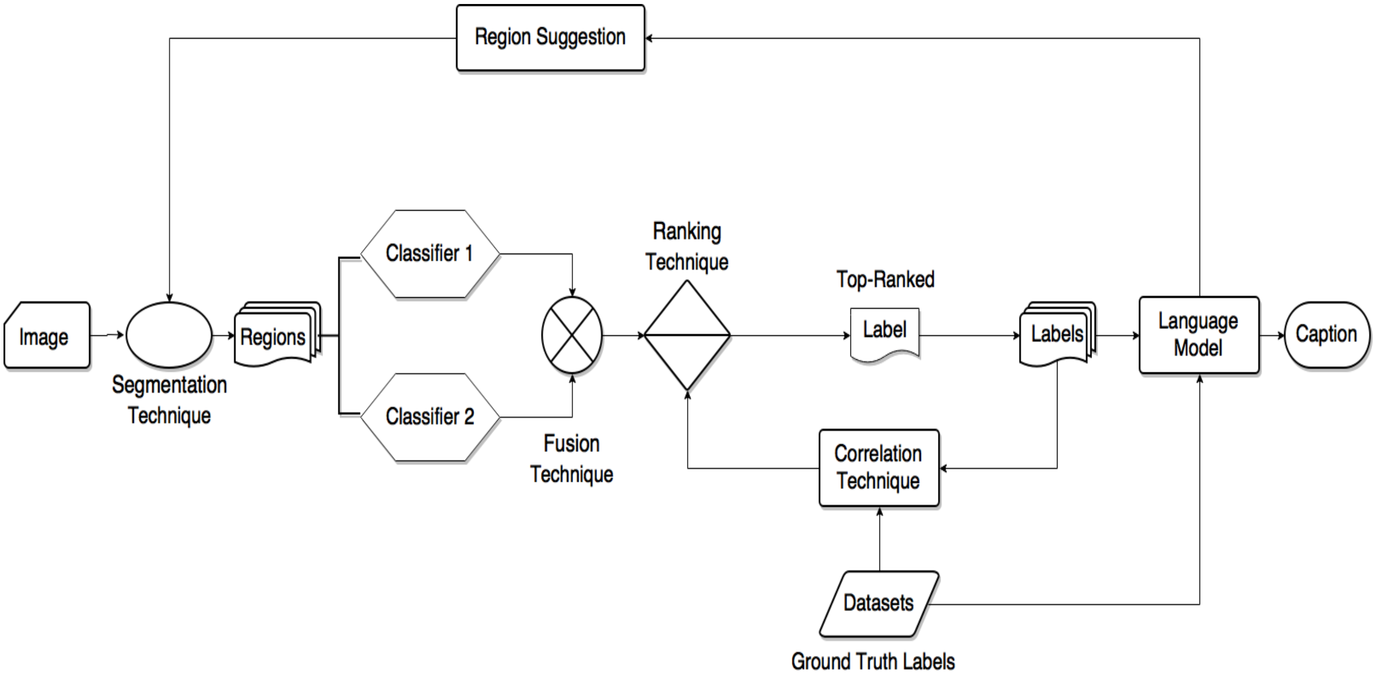


Figure 3.3 Proposed multi-labeled annotation model

In Fig. 3.3, we present the annotation model to label images with a subset of vocabulary keywords. In the first stage, a segmentation technique is used to extract the mean regions in the image. Each region passes through multiple trained classifiers (e.g. FV-HOG and CNN) to find the probability distribution of its label. One of the limitations of current approaches in this domain is that the annotation depends only on the top ranked label of once classifier. Therefore, we need to consider more than recommendation for each region in our approach, for example, we may consider the top *n* suggested labels generated from the classification stage by n classifier for each region. The probability of these labels per region can be different from one classifier to another, therefore, a fusion function is used to decide on the final label such that:

$$F_u\,(\mathcal{L}_{ij}) = \alpha.P\,(\mathcal{L}_{ij}|\,D_1) + \beta.P\,(\mathcal{L}_{ij}|\,D_2) \qquad \forall\,\mathcal{L}\;\in \upsilon \tag{3.1}$$

where α and β are two decision variables that are found during the training phase. $\mathcal{L}_{ij}$ is the jth label suggested based on the classifiers outputs D_1 and D_2 for region i, and υ is the vocabulary which is determined from the words in use by the common datasets.

The classifier fusion process is performed by a ranking technique, in which we may have at least two approaches. First, we can consider the label of the region that has the top ranking as the first label to be generated from the annotation system. Second, we may consider the top ranked labels for all regions, and then consider the one with the highest correlation with others.

The first generated label from the model is used as a seed or first token of the caption subset A_s an initial step, $A_s = \emptyset$ before the seed is created such that $A_s = A_s \cap \mathcal{L}_r$ where $\mathcal{L}_r$ is the label of region *r* that has the highest ranking score among all the labels of the other regions in image *I*.

After generating the first token, a correlation function is used to re-rank the top suggested labels of the other regions such that the label of the other region that has the highest occurrence with the first token label will receive more weight as follows:

$$Fu\,(\mathcal{L}_{ij}) = Fu\,(\mathcal{L}_{ij-1}) \cup \;\; arg\ max\ Corr\,(\mathcal{L}_{ij-1}, \mathcal{L}_{ij}) \tag{3.2}$$

where *Corr* is the correlation function that measures the correlation between the next label; $\mathcal{L}_{ij}$ and the generated one; $\mathcal{L}_{ij-1}$. If there are multiple labels previously generated, we may consider the generated one that has the best correlation value.

The correlation function can be created based on the ground truth captions which are subjectively annotated in common datasets. A co-occurrence matrix is created for all the words of the ground truth captions and used to refine the next suggested label according to the last formula (3.2).

The labels generated with reasonable consistency is passed to a Language Model (LM). The model is used to estimate the probability of a new region label conditioned on the preceding region label generated by the model. This process contributes to the retrieval stage of the annotation model by finding more relevant unrecognized regions. This step makes the retrieval and classification stages benefit from the semantic domain which constitutes one of our main contributions in this dissertation.

The previous approach can be enhanced during the development process by optimizing its parameters to guarantee best possible reliability and accuracy for the annotation system.

4 STATISTICAL BASED IMAGE TAGGING

In this chapter, a statistical based image tagging prototype is introduced. The statistical mage tagging is based on using normalized multidimensional color histograms as a global descriptor of low level features of images is an extension of earlier work [45]. The histogram based concept has been tested using Corel 1K dataset [31] which includes ten categories of images represented by global features. The k-nearest neighbor rule is used to predict the closest tags that can be assigned to untagged images. Tag assignment of the untagged images, involves a statistical prediction method that implements a joint probability distribution to rank the appropriate tags. Despite the simplicity of this basic model, it outperforms most of the learning based methods in terms of accuracy and speed. The main reason stems from the fact that such a simple technique does not require a large number of training examples compared to most of the ML-based techniques.

4.1 Image Retrieval Layer

The first layer of the framework is the CBIR layer, in which the untagged image features are extracted and matched with the nearest image features stored in the feature database.

The accuracy of the tagging model depends on the output from the retrieval technique and the optimization of the tagging parameters. In the KNN Histogram (KH) descriptor, RGB color space is used in computing the feature vector $\boldsymbol{f}$ as shown in Fig. 4.2. The color histogram descriptor algorithm quantizes the color histograms for each channel of the (R,G,B). The Euclidean color histogram $\boldsymbol{f}_H$ vector is computed for each image according to the following equation:

$$\boldsymbol{f}_H = \sqrt{(\boldsymbol{H}_R)^2 + (\boldsymbol{H}_G)^2 + (\boldsymbol{H}_B)^2} \tag{4.1}$$

where $\boldsymbol{H}$ is scale invariant histogram for each RGB channel. The dimension of the vector is controlled by the number of quantization levels which is a critical factor in determining the retrieval accuracy. The number of bins used is 5 which has been determined based on a training phase and verified to be a reasonable value.

The optimal features vector dimension is achieved through an optimization feedback loop controlled by precision factor as shown in the image tagging section. The vector Q in Fig. 4.1 represents the feature vector of the images in the database. The KNN distance is used to rank the closest neighbors between the untagged image histogram feature vector and the database image features vectors.

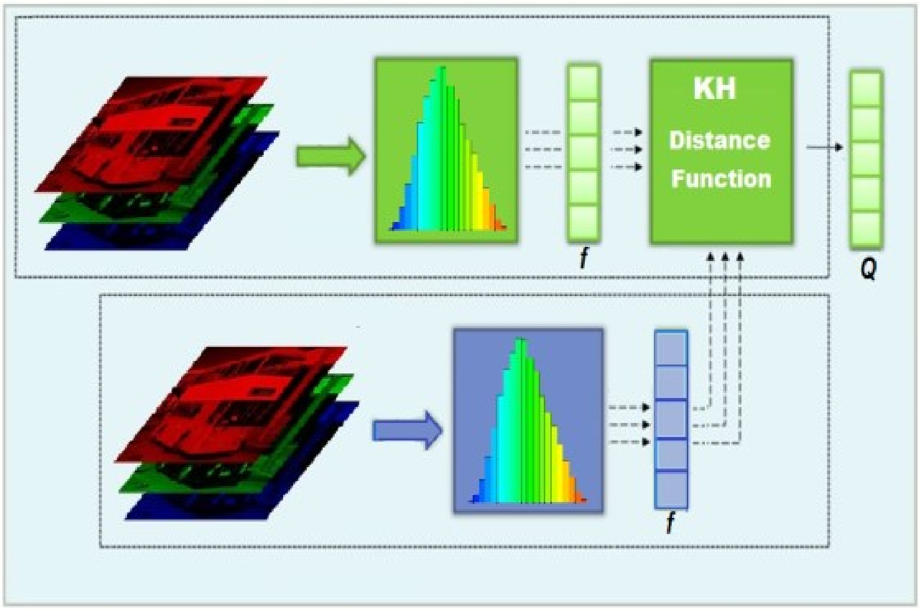


Figure 4.1 KH schematic diagram

4.2 Image Recognition Layer

In this layer, the image database is used as a library to identify the untagged image I according to a predefined image concept. With 10 different predefined concepts C_i such that $i \in \{1,..,10\}$, and 10 retrieved images from the database in the Q vector, the classifier uses all the top ten retrieved neighbors to perform image concept recognition as follows:

$$R(I) = argmax \sum_{i=1}^{m} \Pr(\, Q_i | \boldsymbol{C}) \tag{4.2}$$

where $R(I)$ is the recognized concept for image *I*.

4.3 Image Tagging Layer

As a top layer, auto-tagging aims to annotate a group of untagged images, and retags each image according to the statistical data received from the recognition layer. Because this layer depends on the recognition layer, a training set from the image database can be used here as a library of concepts to optimize the model such that both the retrieval layer and the recognition layer can be optimized to decrease number of the incorrect tags recommended by the top layer. The proposed training block diagram for the framework is shown in Fig. 4.2.

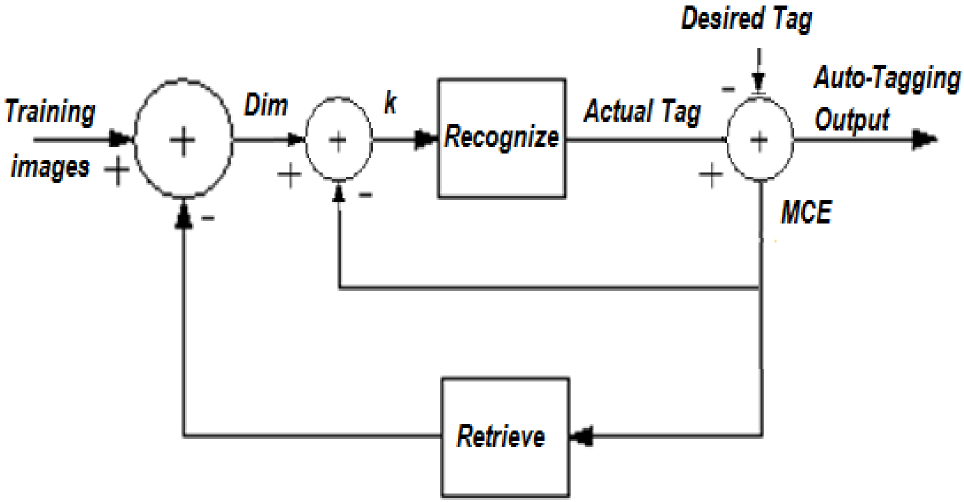


Figure 4.2 Training and optimizing block diagram

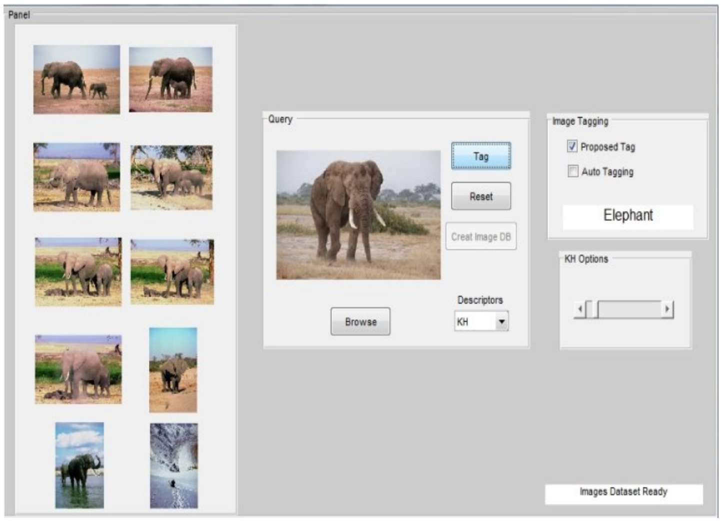
The auto tagging system is probabilistic approach, that is, the more frequent features received for a certain concept, the higher the probability to use this concept as a recommended tag for the particular image.

Basically, in the image dataset, each set of image low level features $\boldsymbol{f}_H$ is associated with a text representing the concept or simply the tag of that image such that $I_x = \{(\boldsymbol{f}_{H_x}, C_x)\}$ for each image, $x \in \{1, .., n\}$ where n is the number of images in the database. Therefore, the image database is used here as a library of concepts to predict the closest tag. The same tag can also be used in the training dataset of the tag prediction model.

4.4 Experiment

Corel-1K image dataset [31] is a popular dataset that we have used for evaluating the automatic tagging. In Corel-1K dataset there are 10 categories each with 100 relevant images. Each category represents a different semantic concept. The proposed auto-tagging multilayer model is coded in Matlab, and a snapshot of the results is shown in Fig. 4.3, where a new untagged image that does not belong to the image dataset and exists on the web is used to evaluate the tagging system.

The results shown in Fig. 4.3 reflect the performance of the KH descriptor. The results are based on the Chebyshev distance measure. The results also show the accuracy of the statistical based approach in tag recommendation. In order to evaluate the efficiency of the retrieval layer, the precision measurement is used to compute the ratio between the number of retrieved relevant images and the total number of retrieved images.



☑ Proposed Tag

□ Auto Tagging

Elephant

Descriptors: KH

Images Dataset Ready

Figure 4.3 Example of auto tagging framework

Table 4.1 Comparison of the Average Retrieval Precision (ARP) rates

Retrieval Method	Average Precision
Block Based LBP [30]	0.230
MSD [27]	0.536
Color, texture, shape [29]	0.549
Image Based SIFT-LBP [32]	0.657
Our KH Method	**0.672**

A comparison between retrieval precision rates of the proposed descriptor and other techniques is given in Table 4.1, which shows that our proposed technique is more suitable for Corel-1K images than other existing techniques. The results also indicate that many categories in Corel 1K dataset are best represented by global descriptors rather than local descriptors (e.g. SIFT) or texture-based descriptors (e.g. LBP). Our proposed system which is referred to by the KH method achieves the highest overall precision rate over existing approaches. The accuracy of the retrieval stage has direct impact on the efficiency of the recognition layer. The higher retrieval precision results in less tagging error.

A comparison between our statistical approach and other machine learning classifiers has been performed. To evaluate our statistical approach performance in predicting image tags, three popular machine learning methods are used: Random Forest (RF), Neural Network (NN), and Decision Tree (D-Tree). RF classifier is an ensemble learning method that operates by constructing a multitude of D-trees to obtain better tagging prediction performance, while NN is a well-known classification technique that is robust for handling noisy data.

The experiment has been setup up for the RF using 500 trees with 2 variables per level. In the NN, a hidden layer of 100 nodes and 10 nodes in the output layer.

For evaluation purposes, Corel-1k dataset is partitioned into two main sets: a training set consists of 700 images, and a testing set of 79 images. The accuracy is measured by computing the ratio between the number of the correctly tagged images and the total number of the tested images. The accuracy of these techniques is shown in Fig. 4.4, where the accuracy of the D-tree, NN, RF, and statistical KNN (KNN-S) are: 0.456, 0.468, 0.57, and 0.671 respectively.

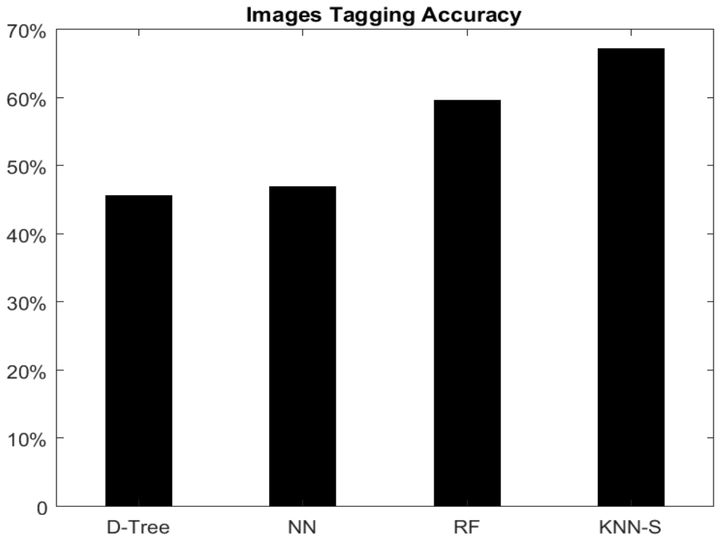


Figure 4.4 Comparison of image tagging accuracy with 10 concepts of Corel-1K

4.5 Summary

In this chapter, we proposed a new image tagging framework that consists of multilayer prediction system. The framework architecture combined with our proposed descriptor (KH) has shown considerable improvement over other comparable technique when tested with the Corel-1K image database. These test results indicate that the global descriptor has high robustness in automatic tagging for these types of images. Our algorithm has been compared to four different techniques. The overall performance of our proposed technique is the highest in both image retrieval and automatic tagging accuracies. The results indicate the suitability of our technique for improving the automatic tagging of images.

5 AUTOMATICALLY GENERATED SEMANTIC TAGS OF TEXTURE IMAGES

Automatic image categorization is the process that classifies an image and categorizes it into a semantic predefined concept based on the image visual content. Classification based on visual contents is a very challenging task due to the diverse variations of objects, shapes, textures and colors that underlying each class. In this chapter, we build a new texture dataset that represents images of artifacts as shown in our work [46]. The dataset is organized into seven categories of 100 images each. The classification pipeline is evaluated with several encoders, and feature extractors. Among the evaluated methods, Integrating Speedup Robust Features (SURF) algorithm and Histograms of Oriented Gradients (HOG) features. SURF-HOG show the best classification accuracy when encoded with IFK.

5.1 Introduction

In the area of CBIR, diversifying image features to include color, texture, and shape, or global and local features proved to be the most common approaches. However, the accuracy of the categorization system doesn’t depend only on the extracted features, but also on the encoding technique, the classifier in use and the tuning of the main parameters of the system pipeline [38].

Most of the traditional feature extraction approaches have many limitations and they are application specific. While color based features are orientation and scale invariant, they do not provide useful spatial information unless we patch the image in multiple regions for better retrieval accuracy. Shape feature descriptors for binary and gray scale images have been used in image retrieval as in the region-based moment descriptors [48], or contour-based (e.g. Fourier-based descriptors). Despite the advantage of each approach and the enhancements achieved in the Fourier-based techniques in terms of time complexity or affine invariant, but still both approaches have multiple restricted requirements and limited number of applications.

In general, the CBIR has many challenges, which makes mid-level features receive more attention in last decade. In keypoint detection and description, SIFT, and SURF [49] are among the most widely used algorithms today for their invariance in translation, scaling, and rotation. However, SURF is considered as a faster version of SIFT and also shows a robust performance in object detection and image classification. The Maximally Stable Extremal Regions (MSER) descriptor [50] is another example of the med-level feature methods. HOG descriptor [51] has shown also good results in face and action recognition. Unlike global descriptors which consider the global image features such as the color histogram, local descriptors breakdown the image into very small regions and consider the properties of these regions (e.g. shape corners) as features for retrieving similar images. In this chapter we are expanding the potential of the med-level descriptor and extend their application to a new dataset of texture images by integrating SURF and HOG features and encoding them with the improved fisher kernel technique to raise the accuracy of the classification model.

5.2 Proposed Categorization Model

The proposed model for the image categorization is based on BoW approach. BoW is a popular local feature based technique for image classification inspired by Bag of Words model that is used in text mining. In document retrieval, the frequency of words in documents are used to classify or retrieve similar documents, the same idea inspired in the image retrieval and classification by using image features as visual words and classify the image based on the histogram of the frequency of visual words. The visual word vocabulary can be generated by clustering the image local features, for example in SURF descriptors of key points are often used as feature vectors. These vectors can be 64 or 128 dimensional gradient based feature vector. By finding all SURF key points in an image and using, for example, K-means clustering, the means of each cluster constitutes a visual word. Therefore, if there are K clusters generated then the dimension of the bag of visual words is K. This approach is also known as codebook representation.

In order to find the k-means clustering for the image, suppose the SURF descriptors of a category is given by n SURF keypoints, and each keypoint is represented by a vector x of dimension 64 such that the features space is $x_1, .., x_n$, and the K-means objective is to find K centers $c_1, .., c_k$ and assignments $q_1, .., q_n \in$ {1,..,K} of the points to the centers such that:

$$\min E(c_1, .., c_k, q_1, .., q_n) = \sum_{i=1}^{n} ||x_i - c_{qi}||_p^p \qquad (5.1)$$

where E is the minimum sum of distances between the point and the centers. Many algorithms are used to estimate these centers which represent the vocabs of the codebook. After forming the codebook, each image is represented by a histogram of the vocabs in the codebook. By training the categorization model, the relation between the histogram of the vocabs and the art categories can be determined. Therefore, this categorization system is a combination between the unsupervised and supervised techniques.

As an alternative approach to the BoW technique, FK is a powerful framework shows competitive performance in the field of image classification. The advantage of this technique is that it combines the strengths of generative and discriminative models. In generative models (e.g. Naïve Bayes, GMM) we are given some sample data and labels where the model finds the hidden parameters and specifies the joint probability distribution between the observed and target variables. For discriminative models (e.g. SVM, Neural Networks) which are also called conditional models, they allow only sampling of the target variables conditional on the observed quantities. Therefore, discriminative models focus directly on the classification problem while generative models can handle variable length data.

The main idea of FK approach in image classification domain is to characterize the input images with a gradient vector derived from a generative probability model which is a GMM. Because GMM is a mixture of K multivariate Gaussian distribution, therefore, if we are given a features space $x_1, .., x_n$ extracted from an image, then $\theta = (\mu_k, \sum_k, \pi_k : 1, .., K)$ is the parameters of the GMM that fits the distribution of the features space such that μ_k and $\sum_k,$ are the mean and covariance of the distribution, π_k is the prior probability of the K.

The GMM can approximate the distribution of the images low level features such as the visual vocabulary. Therefore, the gradient representation here is replaced the histogram of occurrences in the BoW such that:

$$p(x|\theta) = \sum_{k=1}^{K} \pi_k p(x_i| \ \ \mu_k, \sum_k) \tag{5.2}$$

Learning GMM to fit a dataset distributed in a feature space $x_1, .., x_n$ is done by maximizing the log-likelihood of the data such that:

$$l(\theta; x) = \frac{1}{n}\sum_{i=1}^{n} log \sum_{k=1}^{K} \pi_k p(x_i| \quad \mu_k, \sum_k) \tag{5.3}$$

Once the generative model is learned, the representation of each x is determined by its effects on the maximum likelihood parameter estimation such that this effect can be computed as:

$$\varphi(x) = \nabla_\theta log \sum_{k=1}^{K} \pi_k p(x_i| \quad \mu_k, \sum_k) \tag{5.4}$$

Using the gradient vector can help in determining in which direction the model parameter should be modified to best fit the data.

In the context of image classification, the FK can actually be understood to extend the popular BoW by going beyond count statistics. Therefore, we can concisely state that both FK and BoW are based on a visual vocabulary, FK based on GMM while BoW uses K-means clustering, FK properties supports the use of linear classifiers while BoW stipulate nonlinear classifiers to give good classification accuracy. In this chapter we use a special, approximate and improved case of FK called Fisher Vector (FV) which is a statistics capturing the distribution of a set of local image descriptors. We evaluate both of K-means and FV performance with the art dataset as it is shown in experimental section.

A competitive categorization performance in this chapter could be achieved by integrating the FVs of SURF and HOG descriptors. HOG descriptor decomposes the image into patches of squared cells, computes the histogram of oriented gradients in each cell, normalizes the result using a block-wise pattern, and returns a descriptor for each cell. In this chapter, after extracting SURF keypoints, the histograms of oriented gradient around these keypoints are extracted. The GMM of both descriptors features space computed each alone before computing the Principle Component Analysis (PCA) of the FVs. Both of the PCA-FVs are fused for the learning and testing datasets.

5.3 Experimental Results

Matlab and VLFeat [55] are used as the main environments to construct and evaluate the proposed categorization model. The texture dataset is collected from the web and categorized in seven groups as represented in our work [46]. The dataset has 700 images divided into 7 categories each with 100 relevant images and each category is considered as an independent concept. Another 110 images are used to evaluate the categorization model with different techniques and descriptors as shown in Fig. 5.1.

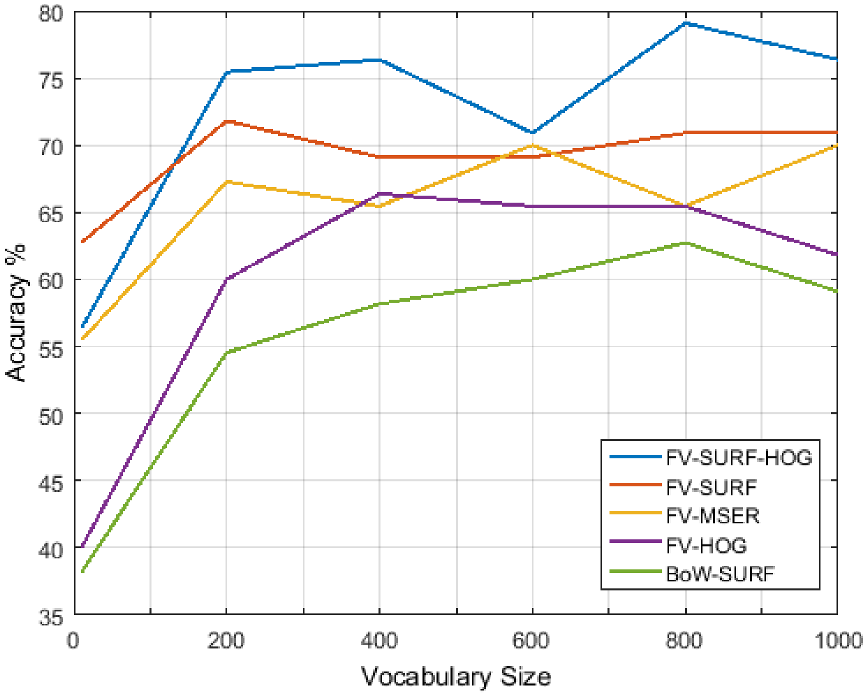


Figure 5.1 Plot of the categorization accuracy of different methods

The testing dataset for the seven different categories is used for the evaluation purpose. The model trained using Support Vector Machine (SVM) as a classifier, the results in Fig. 5.1, show that FV-SURF-HOG method outperforms the other methods for almost all vocabulary sizes.

In Table 5.1, a comparison between the proposed methods shows also that FV-SURF achieves higher accuracy than BoW-SURF by a margin of 9.1% which proves the competitiveness of the FV over the BoW. The performance of the categorization model according to the best accuracy achieved can be considered competitive if we take into account that the dataset has diversity of objects in each category with limited training size of data for each object.

Table 5.1 Comparison of the best categorization accuracy of the methods.

Method	**Accuracy %**
FV-SURF-HOG	**79.1**
FV-SURF	71.8
FV-MSER	70
FV-HOG	66.4
BoW-SURF	62.7

5.4 Summary

In this chapter, we introduced a new dataset of texture images. The dataset has been evaluated against state of the art techniques. We proposed a new image categorization method that minimizes the errors of both SURF and HOG descriptors. This method when encoded with FV outperforms the K-means based Bag of Words' approach. The results also indicate the suitability of our method for the arts` image categorization. Integrating the proposed approach with the Convolution Neural Networks (CNN) is considered and discussed in next chapter.

6 DEEP LEARNING FUSION ALGORITHM FOR TEXTURE IMAGES CATEGORIZATION

The intent of the image classification process is to objectively categorize image visual contents into semantic meanings. The classification process is a challenging task due to the difficulty associated with extracting and identifying relevant shape information. In this chapter, we introduce a new fusion algorithm that combines the strengths of deep learning and mid-level image descriptors. Our approach is evaluated using a newly constructed dataset of texture images as discussed in chapter 5. The dataset is organized into seven categories of 100 images each. The fusion algorithm shows an improvement in the classification accuracy over other state of the art methods.

6.1 Introduction

Trying to automatically classify texture images is a challenging task due to the tinny low level details that can differentiate them from each other. To classify these images we proposed a new fusion algorithm that integrates mid-level descriptors and deep learning. The evaluations of the classification pipeline of the mid-level descriptors involved several encoders, and feature extractors. Among the evaluated methods, Integrating SURF and HOG features for the detected surf point show best artifacts classification accuracy when encoded with the IFK.

In general, traditional feature extraction approaches such as color and shape have many limitations. Color-based extractors for example lack the image spatial information. Shape-based extractors such as region-based moment descriptors or contour-based Fourier-based descriptors have multiple restricted requirements such as the need of segmented regions. Therefore, mid-level features have received more attention as a competitive alternative specifically in object detection and image classification. Among the most notable techniques that are widely used since last decade is SURF descriptor. SURF is considered as a faster version of SIFT, where both of these local descriptors rely on detecting the image points of interest or the so called key-points of location and orientation and use this information for the classification purpose. HOG descriptor is another example of these mid-level descriptors that show also good results in pattern classification such as face detection. While humans are able to classify images based on their global features such as colors or shapes, mid-level descriptor are local descriptors that use local information of regional details (e.g. Corners, gradients) to classify the image. By integrating both SURF and HOG, a competitive classifier could be built that achieves high classification accuracy when encoded with IFK and trained with SVM.

In contrast to the traditional learning approaches, Convolution Neural Network (CNN) represents a different classification method with an end to end learning capability [52]. While the number of training samples play key role in the accuracy of the classification mode, CNN has the advantage of fine-tuning a pre-trained network to achieve high classification accuracy even in the presence of small training set.

In this chapter, we are exploring and expanding the potential of the above mentioned techniques by fusing their results for better classification accuracy, and extending their applications to the dataset of African art images.

6.2 Classification Model

The proposed image classification model is based on two state of the art techniques: The mid-level SURF-HOG descriptor encoded with the IFK technique and the CNN. In document retrieval, the frequency of words in documents is used to classify or retrieve similar documents. Similarly, image features extracted by local descriptors can be used to construct a visual word dictionary that can be used to classify images based on their visual word occurrences [53]. In this chapter, SURF and HOG are used as local descriptors to extract features of artifact images. In SURF descriptor, the key points are often used as feature vectors. These vectors can be 64 or 128 dimensional gradient based feature vector. Therefore, if the SURF descriptor of a category is given by n keypoints, each keypoint is represented by a vector x of dimension 64 such that the feature space is $x_1, .., x_n$.

In order to transform feature space into a visual word dictionary, a powerful encoding technique such as IFK is used. The advantage of this technique is that it combines the strengths of both the generative and discriminative models. The generative model extracts hidden parameters from sample data and selects their labels according to the joint probability distribution between the observed and target variables, while in discriminative model (e.g. SVM) which is also called conditional model; it allows sampling of target variables conditional upon the observed quantities.

In image classification, IFK is used to characterize the input images with a gradient vector derived from a generative probability model which is a GMM Because GMM is a mixture of K multivariate Gaussian distribution, therefore, if we are given a feature space $x_1, .., x_n$ extracted, for example, by SURF then $\theta = (\mu_k, \sum_k, \pi_k : 1, .., K)$ are the parameters of the GMM that fit the distribution of the feature space such that μ_k and $\sum_k$, are the mean and covariance of the distribution, π_k is the prior probability of the K.

Learning GMM that fits a dataset distributed in a feature space $x_1, .., x_n$ is done by maximizing the log-likelihood of the data such that:

$$l(\theta; x) = \frac{1}{n} \sum_{i=1}^{n} log \sum_{k=1}^{K} \pi_k p(x_i| \; \mu_k, \sum\nolimits_k) \tag{6.1}$$

Once the generative model is learned, the representation of each x is determined by its effect on the maximum likelihood parameter estimation such that this effect can be computed as:

$$\varphi(x) = \nabla_\theta log \sum_{k=1}^{K} \pi_k p(x_i| \; \mu_k, \sum\nolimits_k) \tag{6.2}$$

The gradient vector in (6.2) is used to determine the direction in which the GMM parameter should be modified to best fit the feature space. In this chapter we use a special, approximated and improved case of FK called FV which is capable of capturing the statistical distribution of a set of local image descriptors based on the learning GMM. Therefore, there is a need to determine the number of clusters or modes k that the GMM uses. Each of the k modes in the learned GMM is characterized by its mean and covariance. FV uses these parameters to encode each image features into a vector of statistical information. To improve the accuracy of this approach, we integrated the FVs of SURF and HOG features. In HOG descriptor, the histogram of oriented gradients is computed for each patch of squared cells for each image. The computed histogram is normalized using a block-wise pattern. The HOG features around SURF keypoints are also extracted. The GMM for SURF and HOG features is computed for each one. A dimensionality reduction operation is performed using the Principle Component Analysis (PCA) technique for the SURF and HOG FVs. The resulted PCA-FVs are fused and a supervised learning procedure is performed using SVM.

CNN classification is also evaluated using the texture dataset. In CNN, the data has spatial structure such that each layer output in the network $x_l \in \; R^{H_l x W_l x C_l}$ is a 3D array where H and W are the height and width of the data spatial dimensions, and C is the number of feature channels. The main objective of the network is to map an input data x to output vector y using a sequence of layer functions f_l such that $x_l = f_l(x_{l-1}, w_l\ )$, and $y = f(x)$, where y is a vector of probabilities to classify the data input x, and w is the network parameter to tune the CNN during the learning phase. The learning objective is to minimize the loss function $l_y(y)$. While the input of the first layer is an image x, the output is a convolution of x using a bank of linear filters $w_{ij\square k'}$ where i and j are the filter dimensions, k is the number of image channels, and k' is number of filters w. The output of the layer is a k' dimensional feature map such that:

$$y_{i'j'k'} = \sum_{ijk} w_{ijkk'} x_{i+i',j+j',k} \tag{6.3}$$

where $y_{i'j'k'}$ is an intermediate feature map of depth k'. The values of $w_{ijkk'}$ can be selected randomly at first before the network tunes these values during the training phase.

A practical architecture of the CNN that is used in this chapter is the sequence chain of layers. To learn this model, a parameter w_l is computed as follows:

$$\frac{df}{d(vec\ w_l\ )^T} = \frac{d}{d(vec\ w_l\ )^T}[f_l(.\,; w_l)..f_1(x_0; w_1)] \tag{6.4}$$

In order to increase the training speed, nonlinear layers following the convolution ones can be used. A practical and simple function is the ReLU in which the function output $y_{ijk} = \max(0, x_{ijk})$. Another important layer is the pooling layer in which a spatial down-sampling of the feature map can be performed to reduce mainly the computational time of the network. In this chapter, max-pooling is used. All activations in the last layer are fed to a FC layer which passes its outputs to a softmax function to convert these outputs to class probabilities such that:

$$p(c_k = 1|x) = \mathrm{softmax}(\mathrm{f}^{out})_k = \frac{e^{f_k^{out}}}{\sum_j e^{f_j^{out}}} \tag{6.5}$$

In this chapter we integrate the prediction accuracy of the CNN and FV-SURF-HOG by using a classification fusion algorithm to improve the classification accuracy as follows:

Algorithm 6.1: Classification Fusion Algorithm

Input: $\mathrm{D_1}, \mathrm{D_2}, \mathrm{t} \in \mathrm{T}, \mathcal{L}_\mathrm{d}, \mathcal{G}_d, \mathcal{G}_t, \mathcal{S}_1, \mathcal{S}_2$
Output: $\mathcal{F}$

```
$\mathcal{S}_1 = \mathrm{Sort}(\mathrm{D_1}, \mathrm{descend}), \mathcal{S}_2 = \mathrm{Sort}(\mathrm{D_2}, \mathrm{descend}), \mathrm{k} \in \{1,2\}$
For each s, $\mathrm{s} \in \mathcal{S}_k$ do
    $\mathrm{e_s} \leftarrow 0$
  For each d, $\mathrm{d} \in \mathrm{D}_k$ do
    if $\mathrm{Score(d)} \geq \mathrm{s}$ & $\mathcal{L}_d \neq \mathcal{G}_d$
        $\mathrm{e_s} \leftarrow \mathrm{e_s} + 1$
    end
  end
  $\mathrm{R_k(s)} \leftarrow \mathrm{e_s}$
end
$\mathcal{S}_{k_{opt}} = \{ \min(\mathcal{S}_k) \,|\mathrm{R_k(e_s)} = 0, \forall\, \mathrm{s} \in \mathcal{S}_k, \mathrm{k} \in \{1,2\}\}$
For each t, $\mathrm{t} \in \mathrm{T}$ do
    if $\mathcal{L}_{1t} = \mathcal{L}_{2t}$
        $\mathcal{F}(\mathrm{t}) \leftarrow \mathcal{L}_{1t}$
    else if $Score(\mathcal{L}_{1t}) \geq \mathcal{S}_{1_{opt}}$
        $\mathcal{F}(\mathrm{t}) \leftarrow \mathcal{L}_{1t}$
    else if $Score(\mathcal{L}_{2t}) \geq \mathcal{S}_{2_{opt}}$
        $\mathcal{F}(\mathrm{t}) \leftarrow \mathcal{L}_{2t}$
    else
        $\mathcal{F}(\mathrm{t}) \leftarrow \mathcal{L}_{1t}$
    end
end
return $\mathcal{F}$
```

D : training set, $\mathcal{L}$: predicting label, $\mathcal{G}$: ground truth, T: a test set, $\mathcal{S}_k$: prediction score for classifier k, $\mathrm{e_s}$:predict error, $\mathcal{F}$:fusion set

In algorithm 6.1, the training datasets D1 and D2 represent the outputs of the CNN and FV-SURF-HOG. Each dataset represents the classifier prediction labels $\mathcal{L}_d$ for the training set and $\mathcal{L}_t$ for testing sets and their scores $\mathcal{S}_k$, where k is the number of classifiers. The algorithm sort the predication scores $\mathcal{S}_k$ of each classifier k output labels in a descending order and then searches for the optimal prediction score $\mathcal{S}_{k_{opt}}$ of each classifier. The optimal score is the minimum score that achieves zero error classification for all the predicated labels that have prediction scores greater than or equal to the optimal one. If both of the classifiers predict the same label then the algorithm uses it. Otherwise, it uses the predicted label by the classifier that has a label score greater than or equal to its optimal prediction score value. If the labels of both classifiers have prediction scores greater than or equal to their optimal values, the algorithm uses the label of the classifier that has higher classification accuracy in the training datasets. The algorithm has a polynomial time complexity of $\mathrm{O}(n^2)$.

6.3 Experimental Results

Matlab is used as the main environment to construct and evaluate the proposed classification model. VLFeat libraries and MatConvNet toolbox [54] are used to build the FV-SUFR-HOG and CNN classifiers. The dataset is collected from the web and organized in seven categories that represent the as presented in our work [47]. The dataset has 700 training images divided into seven classes each of 100 images. The testing dataset has 110 images. The performances of the classifiers are shown in Fig. 6.1.

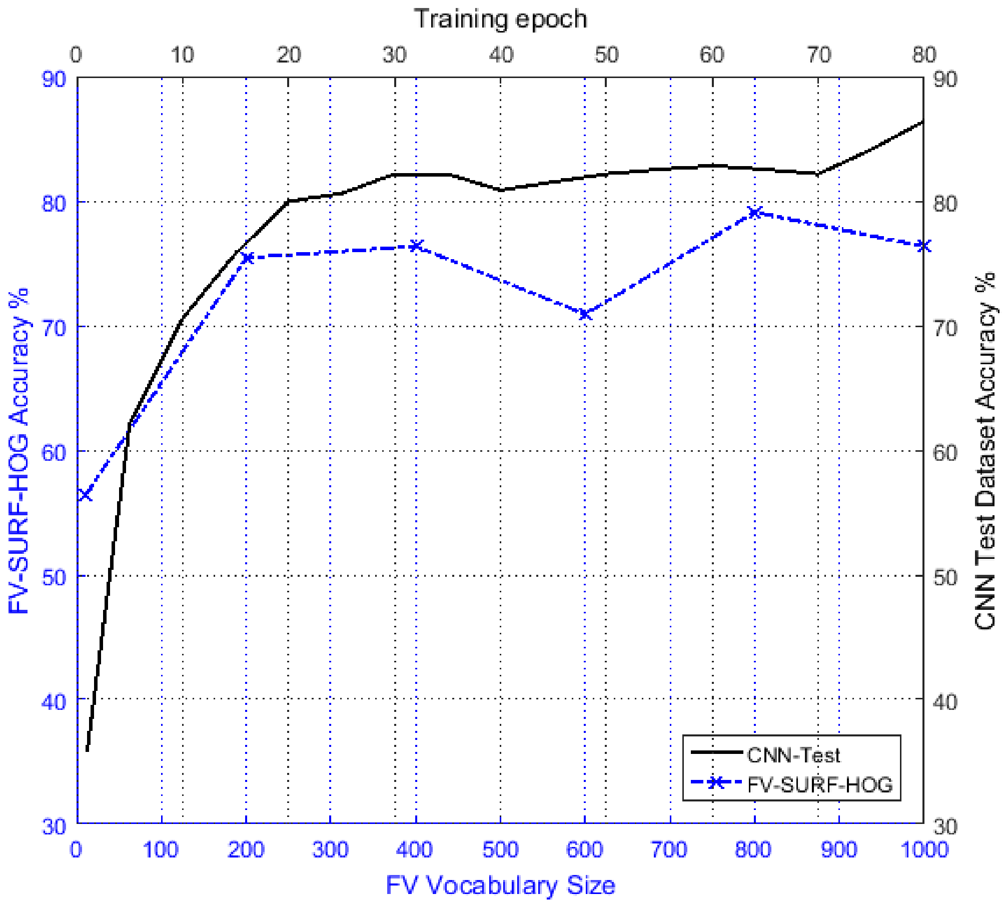


Figure 6.1 The classification accuracy of the fine-tuned CNN and FV-SURF-HOG

We fine-tuned a network that is pre-trained on ImageNet dataset by changing the final convolution layer parameter to fit the seven classes of the texture images. The original network consists of 21 layers. In order to increase the network accuracy, a dropout layer is added before the last convolution layer. The network is fully retrained using the 700 training images. The resulted performance using the 110 testing images is given in Fig. 6.1. The CNN classification accuracy shows the best performance of 87.3% with 80 epoch training, batch size of 50, and learning rate of 10^{-4}. The FV-SURF-HOG classifier is trained using SVM and its results are displayed also in Fig. 6.1. The FV-SURF-HOG method achieves best accuracy at vocabulary size of 800.

To evaluate our algorithm using the test dataset, we divided the test dataset into two groups each of 55 images. The classifiers predicted the scores and labels of each group and the algorithm used the predicted values to find the optimal prediction score of each classifier to predict the labels of the other group. The accuracies of the three methods for the two groups are given in Fig. 6.2.

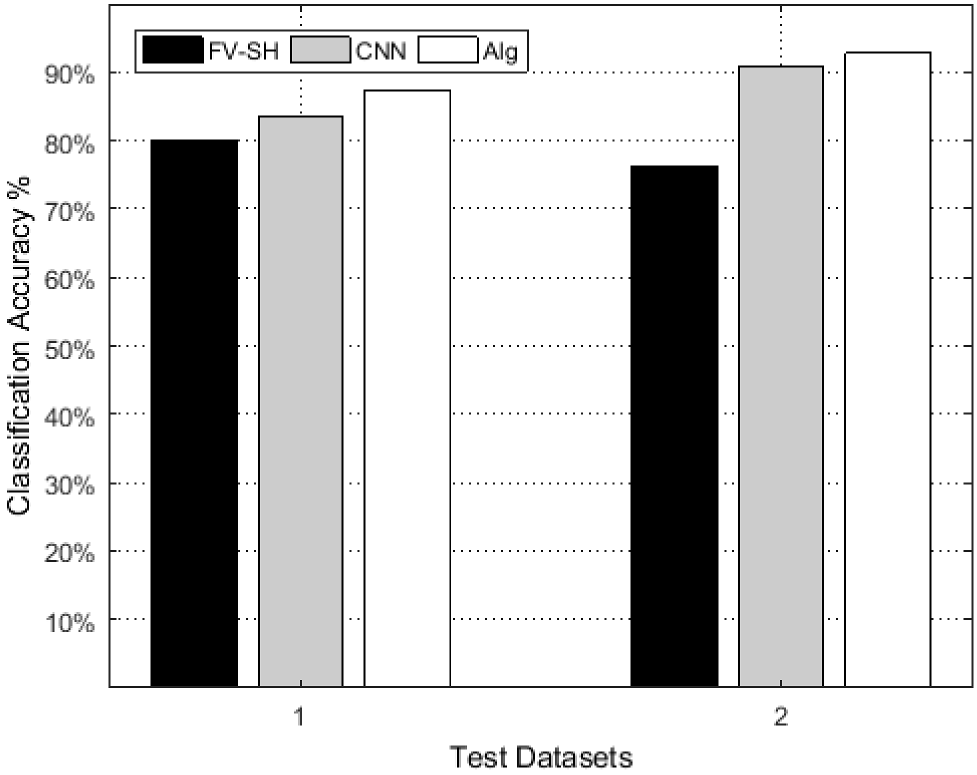


Figure 6.2 Plot of the classification accuracies of the classifiers and the algorithm

A comparison between the proposed methods shows that the fusion algorithm achieves higher accuracy than the CNN and the FV-SURF-HOG which indicates the ability of the proposed fusion algorithm to improve the classification overall accuracy.

6.4 Summary

In this chapter, we introduced a new fusion algorithm that improves the accuracy of the current classification methods. The algorithm is evaluated using a new dataset of texture images. Two state of the art classification techniques, CNN and FV-SURF-HOG, are constructed and used for the fusion process. The results indicate the advantage of our fusion approach in enhancing the CNN and FV-SURF-HOG classification accuracy. Extending the potential of the fusion algorithm to improve the classification of multi-labeled images is discussed in more details next chapter.

7 FUSION ALGORITHM FOR MULTI-LABEL ANNOTATION

In this chapter, we propose a multi-Label annotation system that combines the basic features of fusion techniques with the context information of visual contents. Multiple fusion criteria are selected and incorporated with deep learning techniques as well as mid-level descriptors. Our approach is evaluated using a newly constructed dataset of multi-label images that are organized into thirteen concepts of 50 images each. A comparative study is presented in this chapter that shows an improvement in the classification accuracy of our proposed method against state of the art techniques.

7.1 Introduction

In chapter 2 and 3 we introduced the mid-level descriptors and deep learning for the single labeled texture images, while in this chapter we will evaluate their performance against multi-label images. For the purpose of getting conclusive evaluation, a new dataset for multi-label images classification is used as shown in Fig. 7.1.

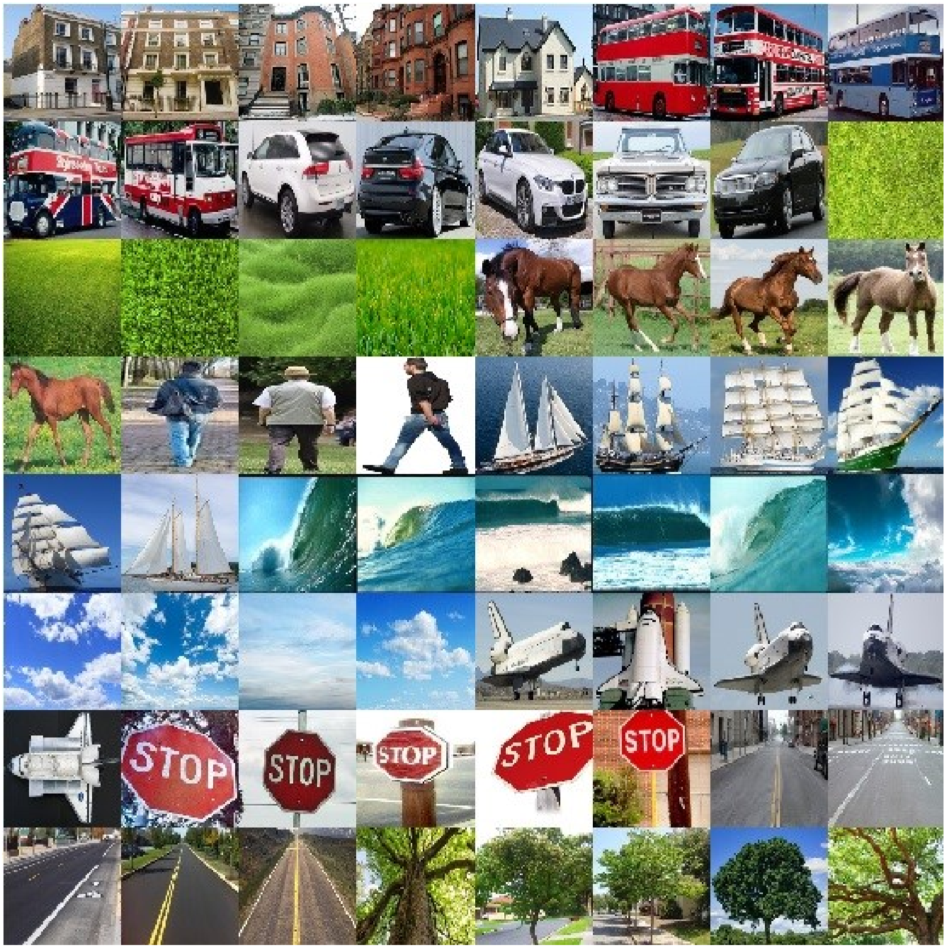


Figure 7.1 Samples of the training dataset

The dataset is collected from the web and organized into 13 categories that represent the following main concepts: {'Building', 'Bus', 'Car', 'Grass', 'Horse', 'People', 'Sailing_boat', 'Sea', 'Sky' , 'Space_Shuttle', 'Stop', 'Street', 'Tree'}; Each concept consists of 50 images used for training the classifiers. The validation dataset has another 130 images to tune the parameters of the classifiers.

Different state of the art classifiers are used to classify different regions in the image. To achieve this task, the semi supervised learning approach is used to train the classifiers. For this purpose, the image regions segmented manually and are fed into the classification model.

Among the mid-level descriptors, SURF and HOG features are evaluated for the classification purpose after encoded with the IFK.

There are 127 image regions that are used for the testing purposes divided into two groups of 62 and 65 regions. The first group of regions is used to test the performance of CNN, FV-HOG, and FV-SURF with different SVM Lambda values as shown in Fig. 7.2.

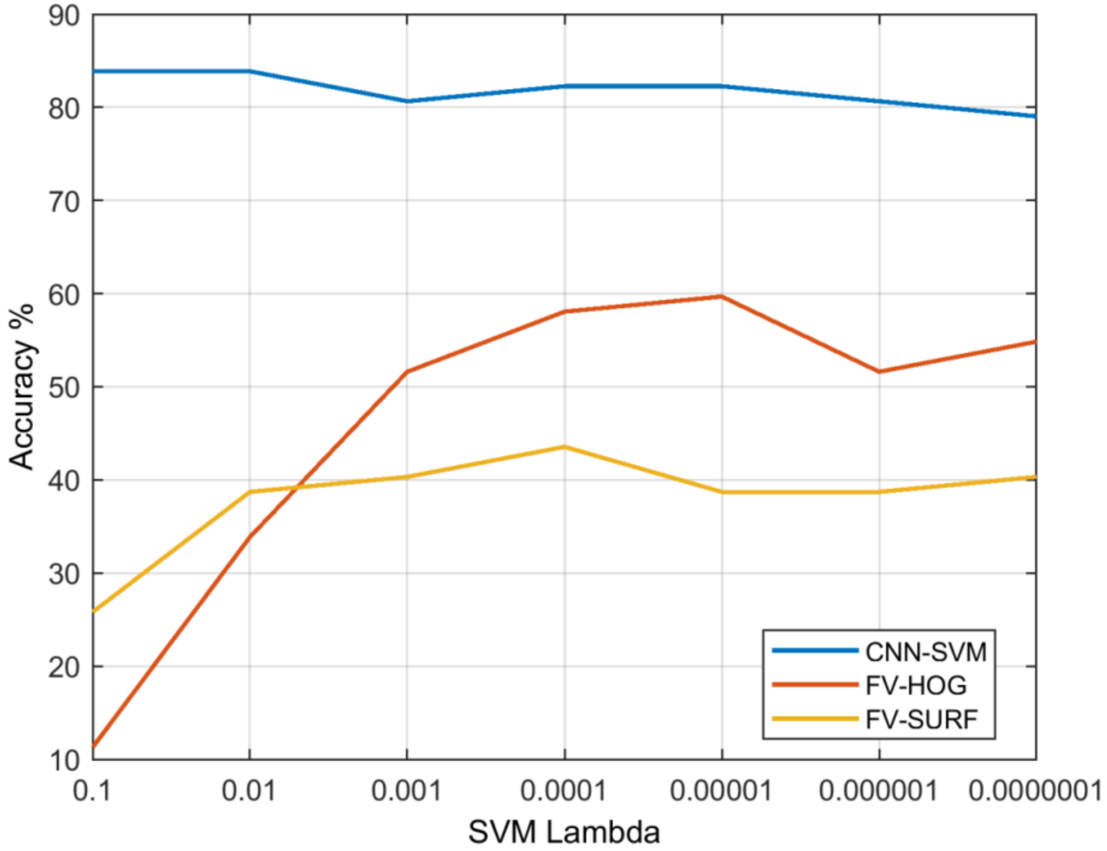
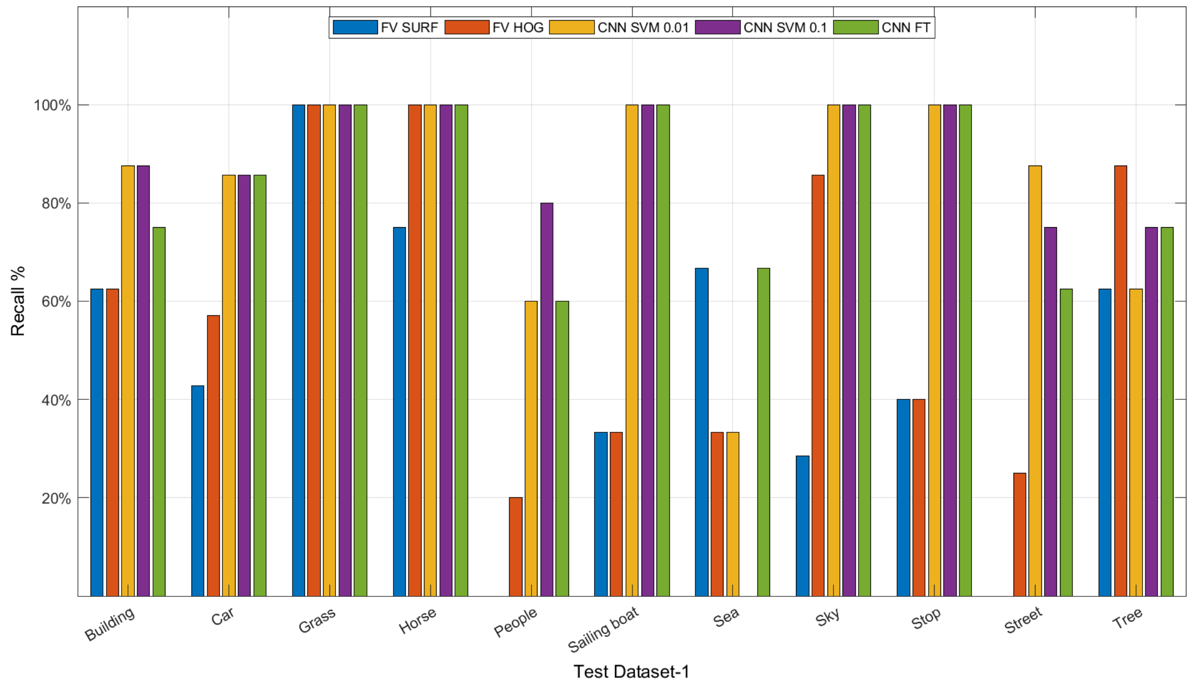


Figure 7.2 Classifiers performance versus SVM lambda

For the CNN, we selected and fine-tuned a network that is pre-trained on ImageNet dataset. The original network consists of 25 layers and it has been trained on more than one million images and it can classify up to 1000 objects. By changing the final convolution layer parameters to fit the 13 classes of our dataset, the network is fully retrained using the 650 training images. After training the network, the outputs of the fully connected layer FC7 are extracted and used as generic features to train the SVM classifier.

The results of the performance of the various techniques are given in Fig. 7.2. The CNN classification accuracy shows the best performance has been achieved by the SVM lambda of 0.1 and 0.01. The FV-HOG and FV-SURF classifiers are trained also using SVM and their results are displayed in Fig. 7.2. The FV-HOG method achieves best accuracy for lambda of 0.00001. Based on the results obtained in Fig. 7.2, CNN with SVM of lambda values of 0.1 and 0.01 are selected in addition to FV-HOG and FV-SURF. Fine-tuned CNN (CNN-FT) is also considered as a reference.

The prediction performance of the various techniques is evaluated on dataset 1 and 2 and is shown in Fig. 7.3 and 7.4. The recall and precision metrics are used as performance measures Given a set of predicted label $\mathcal{L}$ and the ground truth $\mathcal{G}$, let the value of ground truth for that label be $|\mathcal{G}_l|$, and the value of correctly predicted label be $|\mathcal{L}_c|$. The recall can be defined such that; $R = \frac{|\mathcal{L}_c|}{|\mathcal{G}_l|}$, and the precision in Fig. 7.4 can be defined as; $P = \frac{|\mathcal{L}_c|}{|\mathcal{L}|}$. The F-score with a score of 1 for the recall and precision is given in Fig. 7.6. The definitions of the above metrics are given in chapter 2. Also, prediction accuracy as a ratio between the correct predictions and all predictions is also considered.



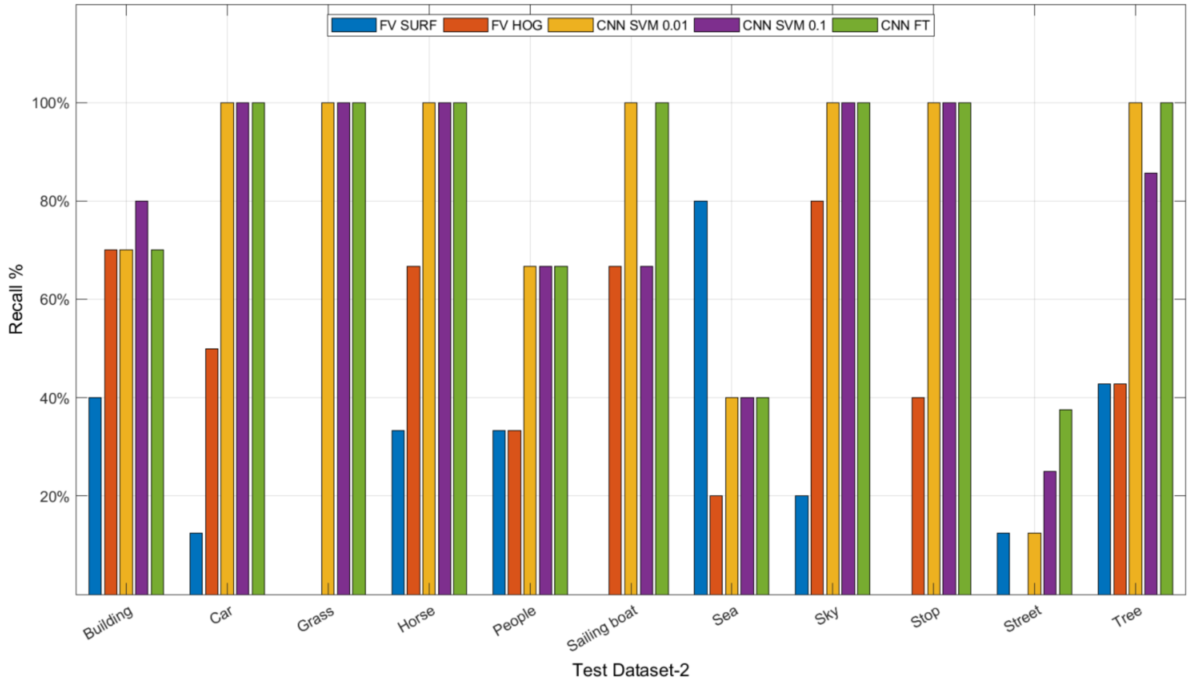
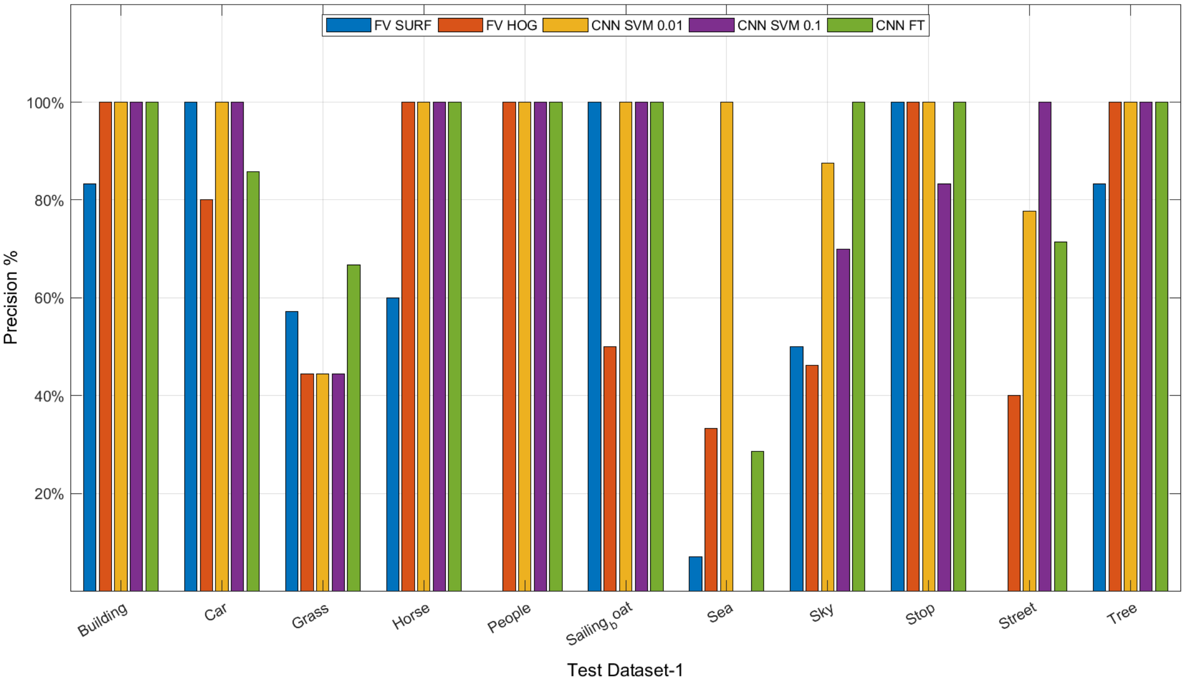


Figure 7.3 Recall metric for the proposed classification methods for each class



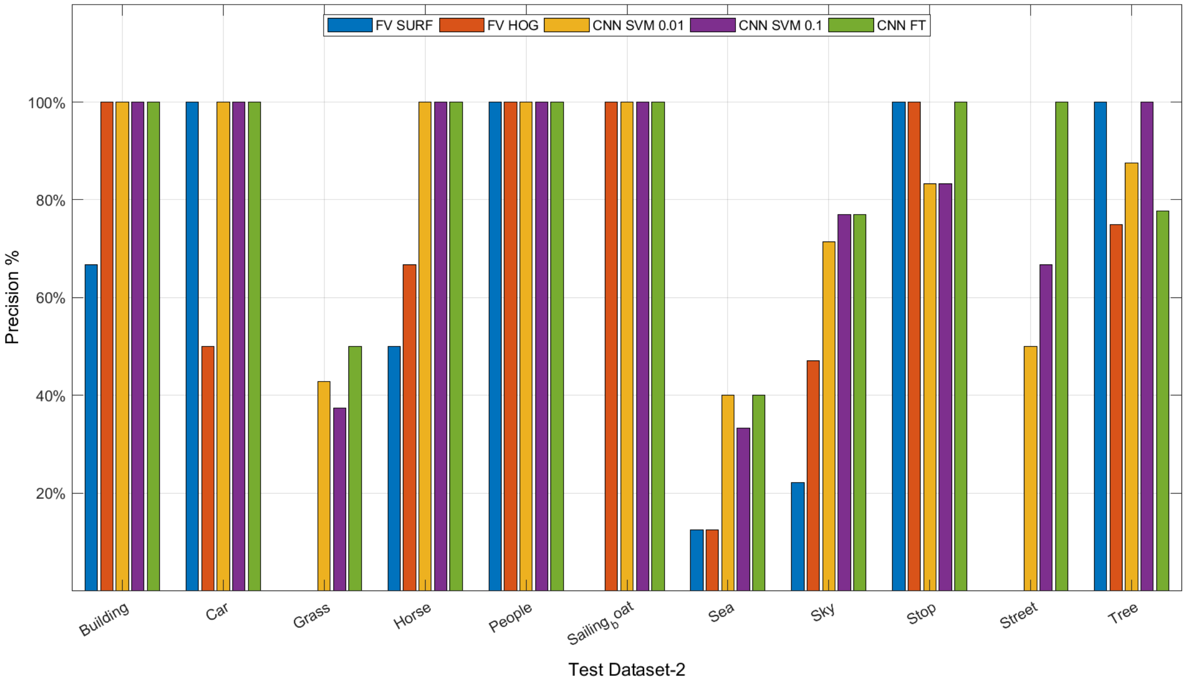


Figure 7.4 Precision metric for the proposed classification methods for each class

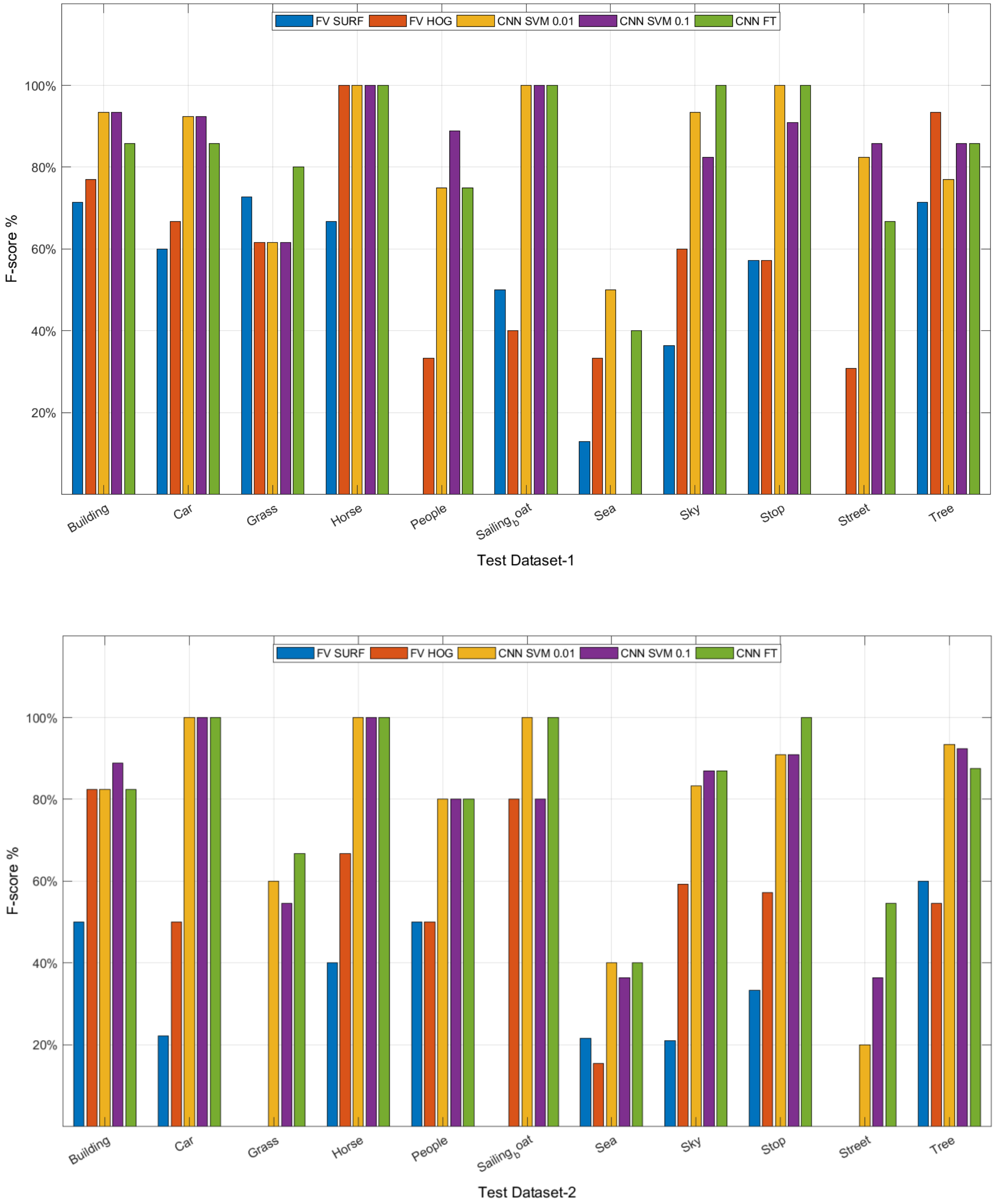


Figure 7.5 F-score for the proposed classification methods for each class

In Fig. 7.4, the precision of CNN varies according to the classification layer in use. The average precision of all classifiers for test 1 and 2 is given in Fig. 7.6.

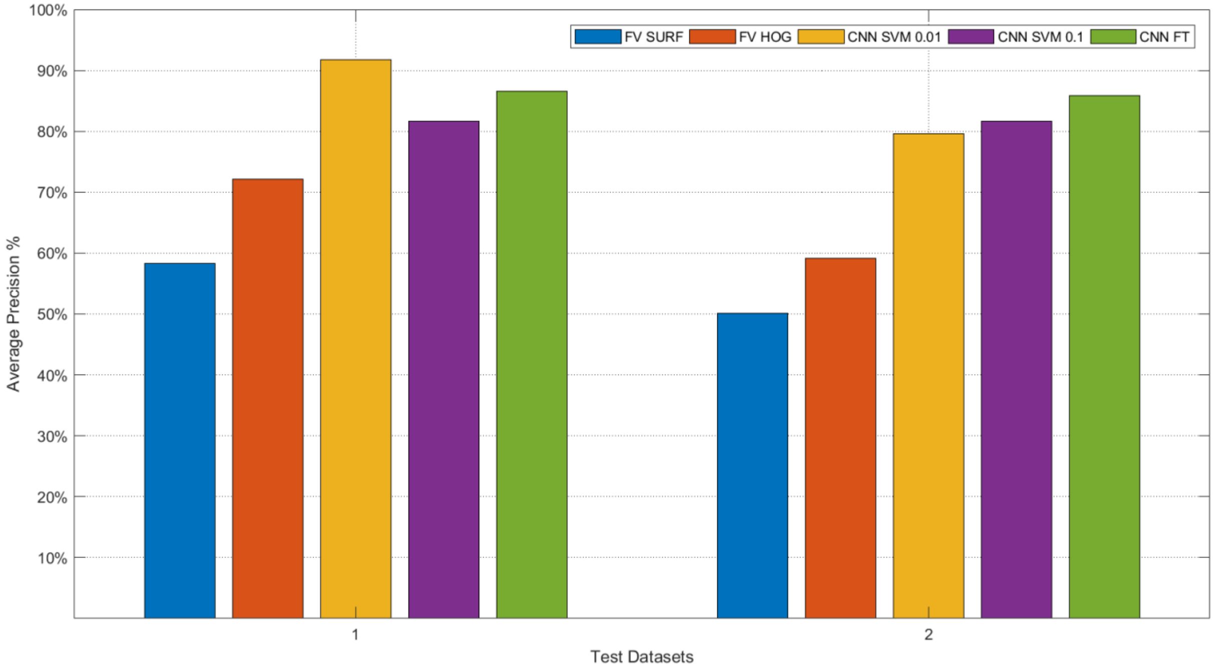


Figure 7.6 Average precision of proposed classification methods for test dataset-1 and 2

In Fig. 7.6, the prediction precision of the two test datasets shows the CNN performance consistency over the mid-level classifiers.

7.2 Fusion Algorithm

In this section we use the classifier fusion concept to integrate the predictions of the several classifiers to improve the prediction accuracy. For this purpose, we propose a prediction threshold value for each classifier to be used as a decision variable in the fusion process. This threshold value can be obtained for each class which we call per_class threshold value, or it can be a global threshold value for the classifier in use. Therefore, we introduce our modified version of fusion algorithm that also incorporates the context information of the visual contents.

Algorithm 7.1: Modified Classification Fusion Algorithm

```
Input: $\mathrm{D}_1, \mathrm{D}_2, \mathrm{t} \in \mathrm{T}, \mathcal{L}_{\mathrm{d}}, \mathcal{G}_d, \mathcal{G}_t, \mathcal{S}_1, \mathcal{S}_2, n$
Output: $\mathcal{F}$
1.  $\mathrm{C}_{lk} = \mathrm{Group}(\mathrm{D}_l), l \in \{1,2\},\ k \in \{1,..,n\}$
2.  $\mathcal{S}_{lk_{opt}} = \ Threshold\ (\mathrm{C}_{lk})$
3.  For each t, $\mathrm{t} \in \mathrm{T}$ do
4.        if $\mathcal{L}_{1t} = \mathcal{L}_{2t}$ & ( $Score(\mathcal{L}_{1t}) \geq \mathcal{S}_{1t_{opt}} \parallel Score(\mathcal{L}_{2t}) \geq \mathcal{S}_{2t_{opt}}$ )
5.          $\mathcal{F}(\mathrm{t}) \leftarrow \mathcal{L}_{1t}$
6.       else if $Score(\mathcal{L}_{1t}) \geq \mathcal{S}_{1t_{opt}}$ & $Score(\mathcal{L}_{2t}) \geq \mathcal{S}_{2t_{opt}}$
7.                if $\mathrm{Metric}_1 > \mathrm{Metric}_2$
8.                    $\mathcal{F}(\mathrm{t}) \leftarrow \mathcal{L}_{1t}$
9.               else
10.                   $\mathcal{F}(\mathrm{t}) \leftarrow \mathcal{L}_{2t}$
11.              end
12.           else   if $Score(\mathcal{L}_{1t}) \geq \mathcal{S}_{1t_{opt}}$
13.                      $\mathcal{F}(\mathrm{t}) \leftarrow \mathcal{L}_{1t}$
14.                  else if $Score(\mathcal{L}_{2t}) \geq \mathcal{S}_{2t_{opt}}$
15.                           $\mathcal{F}(\mathrm{t}) \leftarrow \mathcal{L}_{2t}$
16.                      else if $\mathrm{Metric}_1 > \mathrm{Metric}_2$
17.                                $\mathcal{F}(\mathrm{t}) \leftarrow \mathcal{L}_{1t}$
18.                          else
19.                                $\mathcal{F}(\mathrm{t}) \leftarrow \mathcal{L}_{2t}$
20.                          end
21.                    end
22.                 end
23.           end
24.      end
25.  end
return $\mathcal{F}$
```

D : training set, C : training concepts, $\mathcal{L}$: predicting label, $\mathcal{G}$: ground truth, T: a test set, $\mathcal{S}_{lk}$: prediction score of concept k & classifier l, e_{s} :predict error, $\mathcal{F}$:fusion set, n: the number of training concepts.

In algorithm 7.1, the fusion algorithm training datasets D1 and D2 represent the predicted labels and scores of any two selected classifiers as in Fig. 7.6. In line 1, the predicted labels $\mathcal{L}_{ld}$ and labels score $\mathcal{S}_{lk}$ of each classifier l are organized into groups' C_{lk} where k is the concept that the group belongs to. To find the optimal threshold value $\mathcal{S}_{lk_{opt}}$ of each group of labels k for classifier l we call the threshold function as in line 2. We name this approach a per_class threshold. In case of global classifiers threshold values are used rather than the per_class threshold then line 1 will be modified such that $\mathrm{C}_{lk} = \mathrm{D}_l$ for $, l \in \{1,2\},\ k \in \{1, .., n\}$. The function $Threshold$ is given in algorithm 7.2. The function is designed to find the optimal threshold value of each classifier or for each group of concepts predicted by a classifier.

For evaluating our fusion technique, the test dataset is divided into two groups of 62 and 65 images regions. The classifiers predict the scores and labels of one group and the algorithm uses the predicted values to find the optimal prediction scores of each classifier and used them to predict the labels of the other testing group.

Algorithm 7.2: Prediction Threshold

```
Input: $D,\ \mathcal{L},\ \mathcal{G}$
Output: $\mathcal{S}_{opt}$
Procedure: $Threshold\ (D)$
1.  $\mathcal{S} = \mathrm{Sort}(D, \mathrm{descend})$
2.  For each s, $s \in \mathcal{S}$ do
3.      $\mathrm{e_s} \leftarrow 0$
4.      For each $d$, $d \in D$ do
5.          if $\mathrm{Score}(d) \geq \mathrm{s}\ \&\ \mathcal{L} \neq \mathcal{G}$
6.              $\mathrm{e_s} \leftarrow \mathrm{e_s} + 1$
7.          end
8.      end
9.      $\mathrm{R_k(s)} \leftarrow \mathrm{e_s}$
10. end
11. $\mathcal{S}_{opt} = \{\min(\mathcal{S})\ |\mathrm{R_k(e_s)} = 0, \forall\ \mathrm{s}\ \in \mathcal{S}\}$
return  $\mathcal{S}_{opt}$
End Procedure
```

D : training set, $\mathcal{L}$: predicting label, $\mathcal{G}$: ground truth, $\mathcal{S}$: prediction score, $\mathrm{e_s}$: prediction error, $\mathcal{S}_{opt}$: optimal threshold

The $Threshold$ algorithm 7.2 sorts in a descending order the predication scores S of the predicted labels of each classifier l and then searches for the optimal prediction score S_{opt} of the input labels. The optimal score is the minimum score that achieves zero error classification for all the predicated labels that have prediction scores greater than or equal to the optimal one.

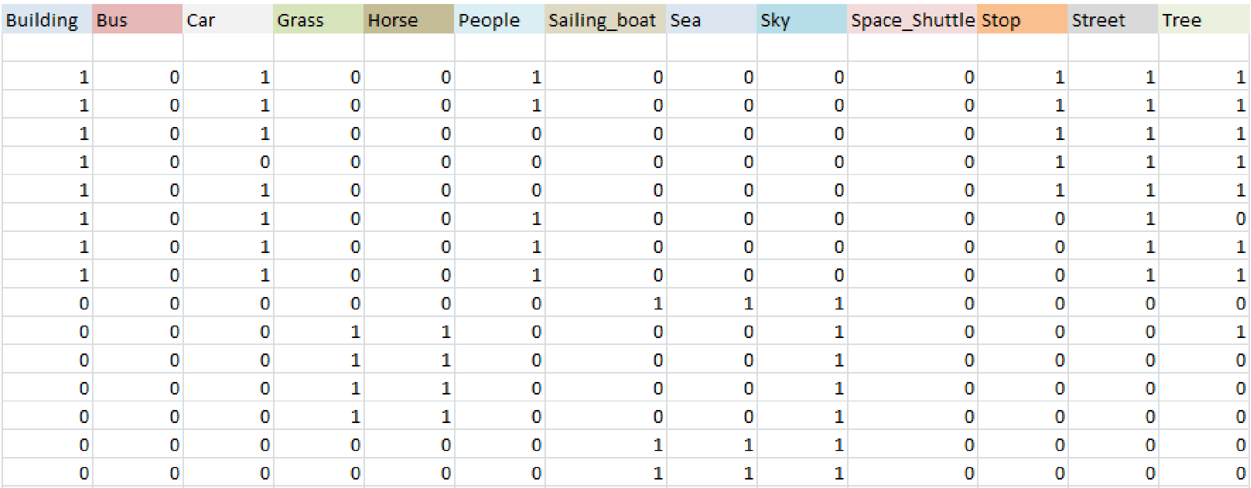
In algorithm 7.1, after finding the optimal threshold values for each classifier, if both of the classifiers predict the same label for the same testing region and one of the classifiers has a predicted value greater than or equal to the threshold value of the predicted class, then the algorithm adds that label to the fusion output. Otherwise, if the predicted labels of both classifiers are not the same, the algorithm checks the classifiers prediction scores. If both scores are above the threshold values, the algorithm uses a matric to compare between the classifiers predicted label and chooses one of them. Different metrics are proposed for the fusion algorithm to increase the fusion accuracy.

7.2.1 Fusion Metrics

In order to increase the precision of our algorithm, we designed two metrics: The accuracy metric, and the word dependency metric.

Accuracy metric: This metric is based on finding the classifier accuracy for the predicted label as shown in Fig. 7.4. If a classifier has a higher precision value off the predicted label, this label is chosen for the fusion output. If class prediction accuracy is unknown for a classifier during the training phase, the average precision value of the classifier can be used in this case as shown in Fig. 7.6.

Dependency metric: This metric is based on finding the word dependency such that the predicted label of the classifier that has the higher dependency on the previous labels will be considered by the fusion algorithm. In order to make this option applicable, a correlation technique is constructed based on the co-occurrence matrices of the testing groups as shown in Fig. 7.7.



Building	Bus	Car	Grass	Horse	People	Sailing_boat	Sea	Sky	Space_Shuttle	Stop	Street	Tree
1	0	1	0	0	1	0	0	0	0	1	1	1
1	0	1	0	0	1	0	0	0	0	1	1	1
1	0	1	0	0	0	0	0	0	0	1	1	1
1	0	0	0	0	0	0	0	0	0	1	1	1
1	0	1	0	0	0	0	0	0	0	1	1	1
1	0	1	0	0	1	0	0	0	0	0	1	0
1	0	1	0	0	1	0	0	0	0	0	1	1
1	0	1	0	0	1	0	0	0	0	0	1	1
0	0	0	0	0	0	1	1	1	0	0	0	0
0	0	0	1	1	0	0	0	1	0	0	0	1
0	0	0	1	1	0	0	0	1	0	0	0	0
0	0	0	1	1	0	0	0	1	0	0	0	0
0	0	0	1	1	0	0	0	1	0	0	0	0
0	0	0	0	0	0	1	1	1	0	0	0	0
0	0	0	0	0	0	1	1	1	0	0	0	0

Figure 7.7 Sample of the co-occurrence matrix of test dataset-1

In Fig. 7.7, each row represents image labels relative to the ground truth. The first one represents existence of this label within the image regions. The co-occurrence matrix is used to find the correlation of main concepts of the dataset. Once a new image region label is recommended by the classifiers the algorithm also considers the new label dependency to be added to the previous accepted labels of the image such that:

$$\mathrm{S}_{cor}(\mathcal{L}_{lt}) = arg\ max\ Corr\,(\mathcal{F}' , \mathcal{L}_{lt}) \qquad (7.1)$$

where $\mathcal{F}'$ is a set of the generated labels for image I that are accepted by the fusion algorithm, $\mathcal{L}_{lt}$ is the recommended label by classifier l for the new testing region. *Corr* is the correlation function that measures the dependency between the new recommended label $\mathcal{L}_{lt}$ by classifier l and the generated ones in the fusion set $\mathcal{F}'$. The correlation can be represented by a conditional probability such that $Corr\ (\mathcal{L}_{lt}\ ,\ \mathcal{F}'\ ) \Leftrightarrow P\ (\mathcal{L}_{lt}\ |\ \mathcal{F}'\ )$. The resulted value S_{cor} is the correlation score of the new test label. If there are previously generated multiple labels, then we may consider only the one that has the best correlation value. Therefore, the new test region label t can be selected such that:

$$\mathcal{F}(t) \leftarrow (\, \mathrm{S}_{cor}(\mathcal{L}_{1t}) > \mathrm{S}_{cor}(\mathcal{L}_{2t}) \;?\; \mathcal{L}_{1t} \colon \mathcal{L}_{2t}\,) \tag{7.2}$$

The overall computational time of the fusion algorithm after incorporating the proposed metrics is low. The algorithm still has a polynomial time complexity of $O(n^2)$. Embedding these metrics with the fusion algorithm shows improvement in performance.

7.2.2 Global Threshold and Dependency Metric

In this section, a case study is presented to illustrate the concepts of global threshold values and dependency metric. The global threshold values of the proposed classifiers for test dataset-1 are shown in Fig. 7.8.

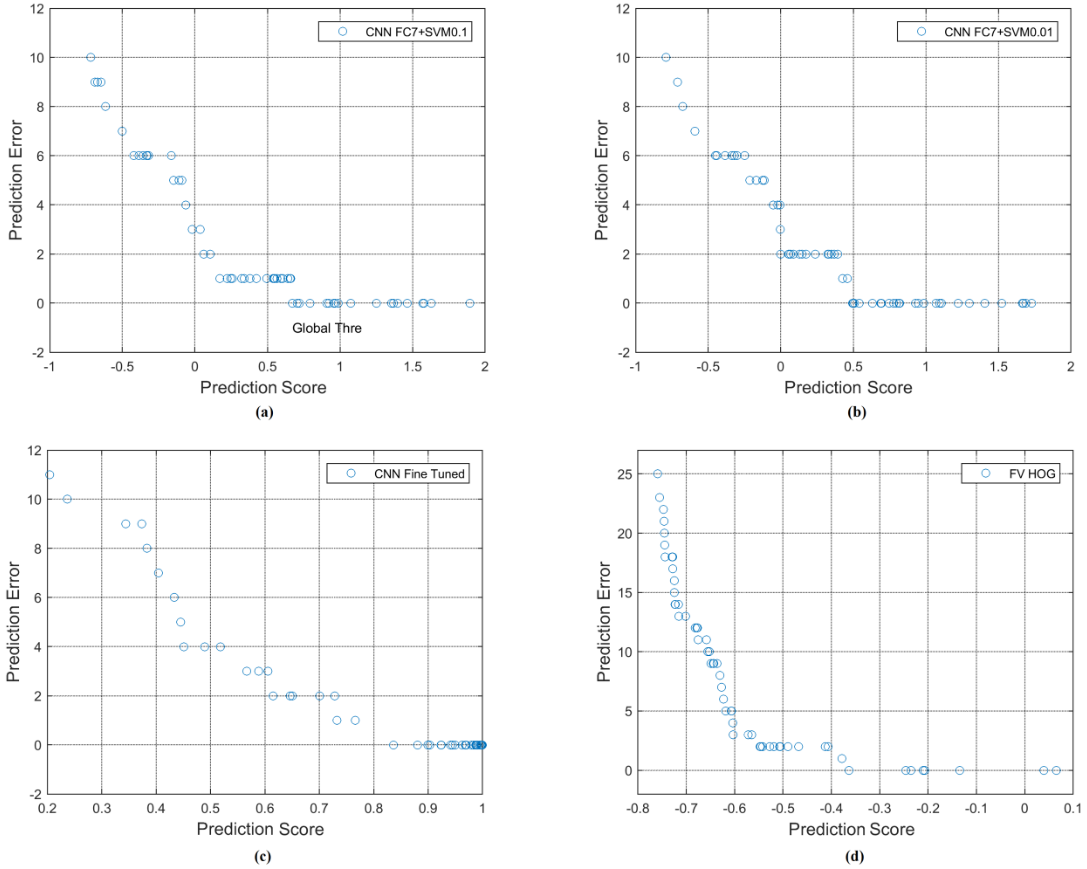


Figure 7.8 Global threshold values of proposed classifiers for test dataset-1

In Fig. 7.8, the prediction scores of the classifiers have different ranges. For example, while CNN-FT has a score range $0.2 \leq \mathcal{S}_c \leq 1$, FV-HOG has a score range $-0.8 < \mathcal{S}_d < 0.1$. Therefore, the global threshold values of the proposed classifiers may have positive or negative values depending on the classifier in use. For that, the design of the fusion algorithm avoids making comparisons between the threshold values. Instead of that, it uses only these global threshold values to decide on the predicted label in two possible scenarios. The first scenario is when the two classifiers have the same predicted label and at least one of the classifiers has a prediction score above or equal its global threshold value. The second one is when the classifiers predicted labels are different and only one of the classifiers has a predicted score that is above or equal to its global threshold value. Otherwise the decision variables in the fusion algorithm are based on the metric in use.

Another approach that can replace the global threshold method is to use a class based threshold values for each classifier as in algorithm 7.1. During the training phase the algorithm uses the predicted labels and scores of the training dataset to find these thresholds. The algorithm uses these predicted labels to organize the labels into groups of same concept and find the optimal threshold value of each group. We call this approach the "Per_Class" or simply the "Class" threshold approach. The performance of the fusion algorithm with different classifiers, different combinations of threshold approaches and metrics is shown in Fig. 7.9-7.11. The best four classifiers in Fig. 7.6 are selected to evaluate the fusion algorithm with different metrics. In three experiments, two of these classifiers are used in each experiment such that CNN-FT is one of them. The average of the results of these experiments show the reliability, consistency, and competitively of the proposed approaches.

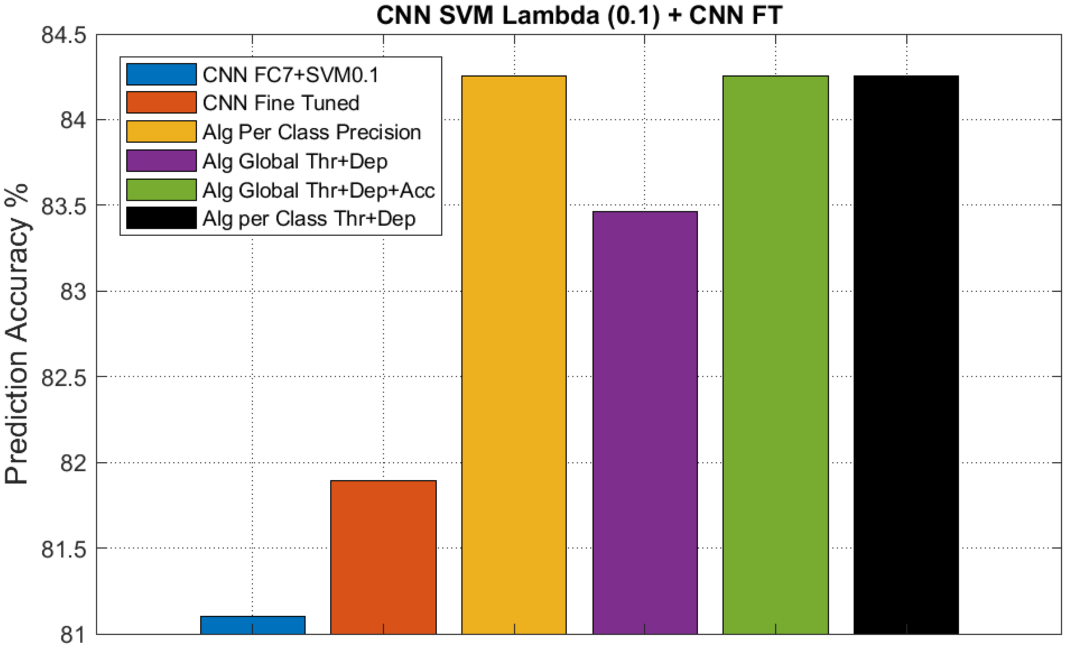


Figure 7.9 Performance of fusion algorithm with FC SVM 0.1

In Fig. 7.9, the algorithm fuses the results of the two classifiers CNN FC7+SVM0.1 and CNN-FT. The performance of each classifier alone is shown. Four different combinations of the threshold approaches (i.e. global, and class) and metrics approaches (i.e. accuracy, dependency, and both) are used.

In general, the fusion algorithm outperforms the stand alone classifiers accuracy. Also, incorporating the class threshold and dependency metric outperforms the global threshold and dependency. Therefore, the class threshold approach shows more consistency than the global threshold approach. However, in order to verify the reliability of the proposed fusion techniques, a k-fold cross-validation technique is used such that the two sets of training labels and testing labels are swapped and the average results are illustrated in the figures.

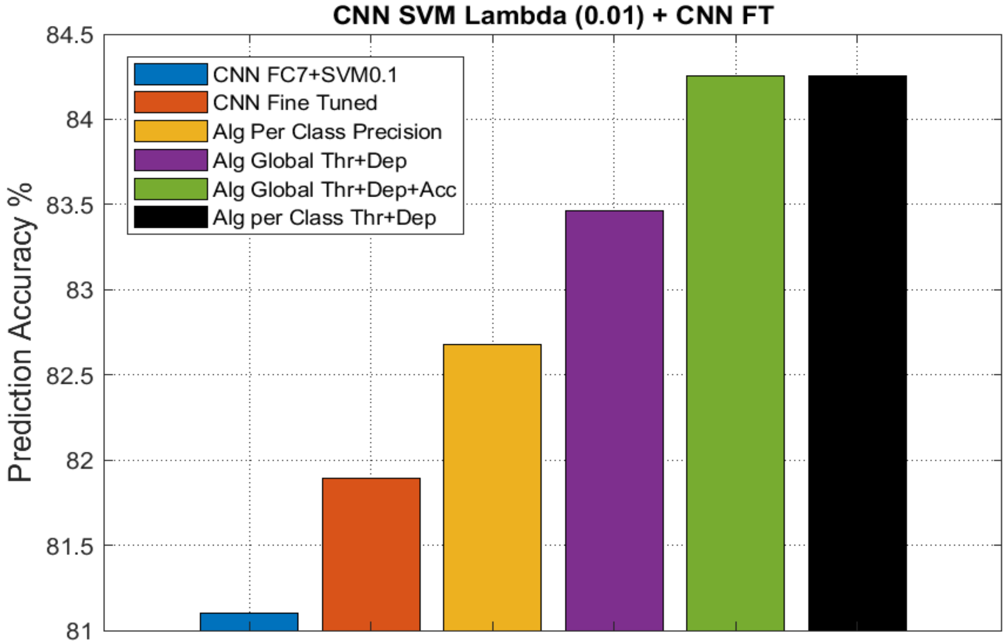


Figure 7.10 Performance of fusion algorithm with FC SVM 0.01

In Fig.7.10, the algorithm fuses the results of the classifier CNN FC7+SVM0.01 with CNN-FT. Again, the fusion algorithm shows an improvement in the prediction accuracy over standalone classifiers. Still the class threshold and dependency incorporation gives best results.

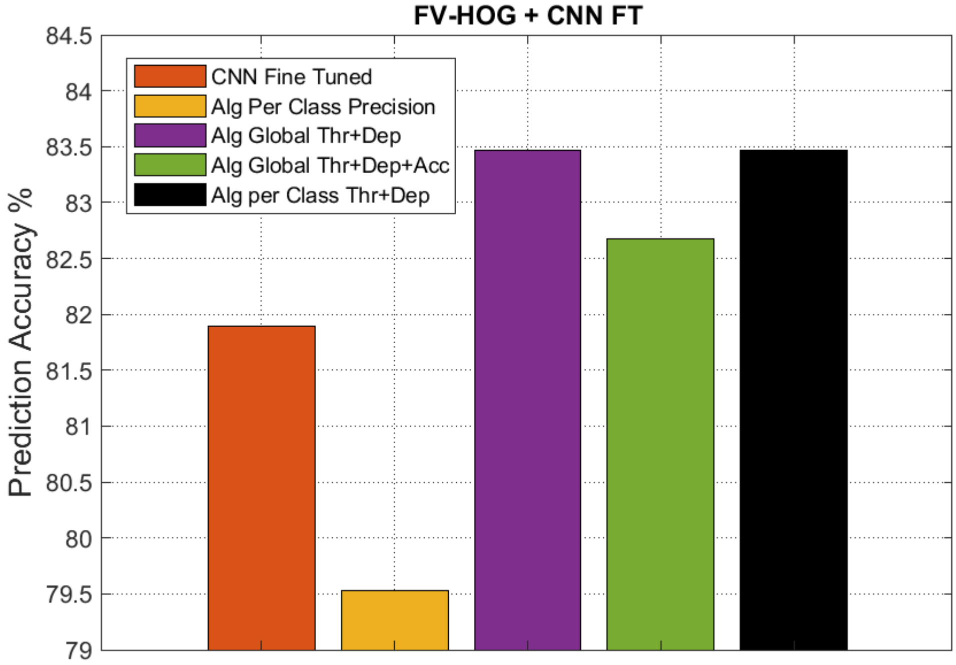


Figure 7.11 Performance of fusion algorithm with FV-HOG

In Fig. 7.11, the algorithm fuses the results of classifier FV-HOG with CNN-FT. The classifier FV-HOG can be considered as a weak classifier compared to CNN techniques. Therefore, this classifier represents a wide range of descriptors that exist today and cannot outperform CNN based approaches. For that, the performance of the fusion algorithm in this case of study is important to verify the possibility of improving the accuracy of CNN by weaker classifiers. From the results, incorporating class threshold and dependency still gives best results. In order to summarize the results obtained so far, the average precision of all methods are illustrated in Fig. 7.12.

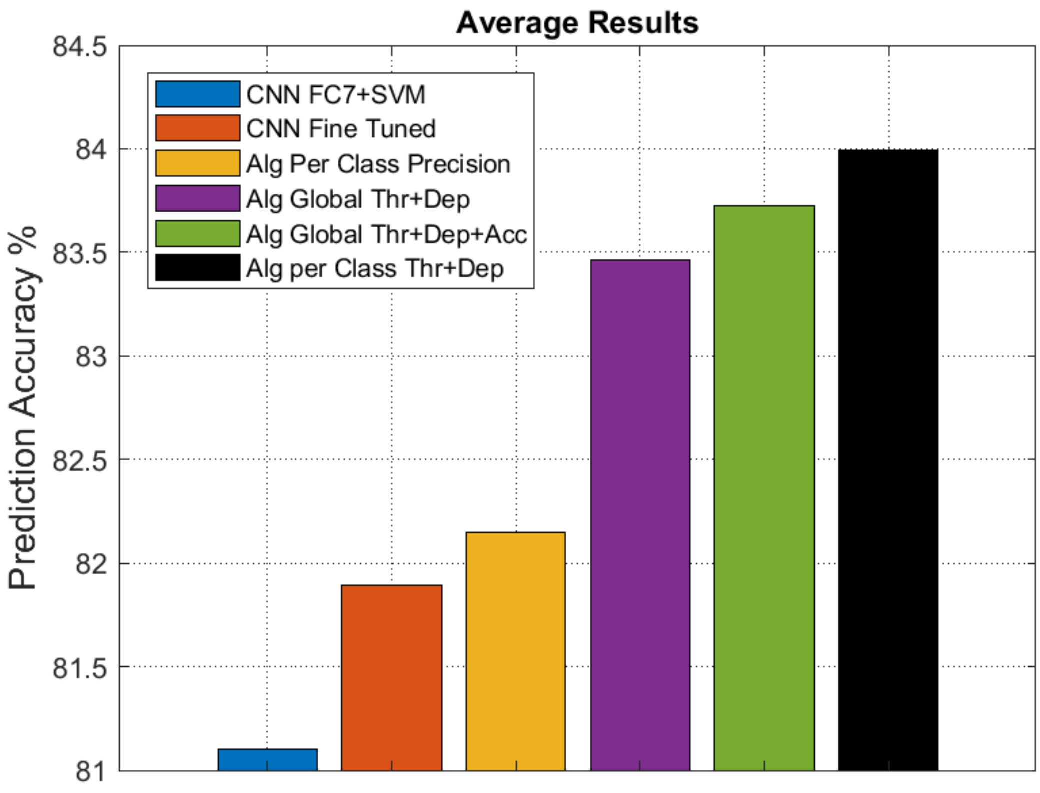


Figure 7.12 Average precision of fusion algorithm

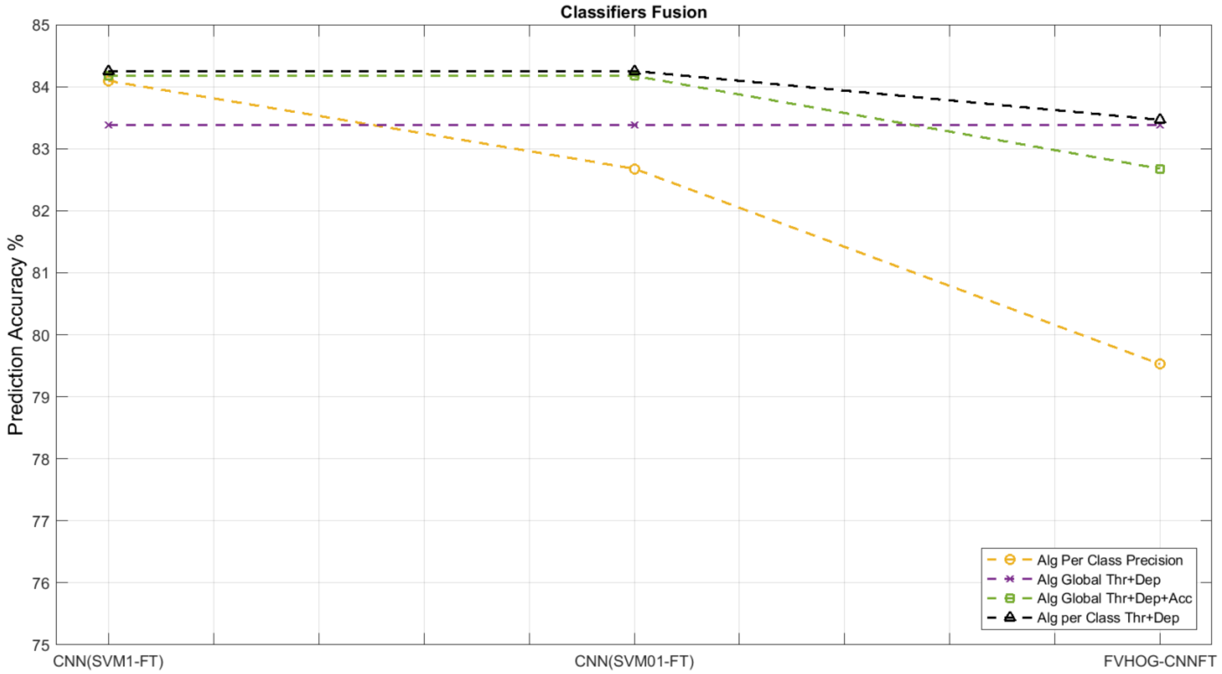


Figure 7.13 Fusion metrics performance versus proposed classifiers

In Fig. 7.12, the average results of all previous experiments shows that incorporating class threshold and dependency outperforms the other approaches including the results of each classifier alone. The consistencies of the proposed fusion metrics is given in Fig. 7.13. The figure shows that the class threshold and dependency metric is more robust and stable comparing with other metrics.

Based on the previous results, it is proved that the fusion algorithm achieves higher accuracy than the CNN and the FV-HOG. This indicates the capability of the proposed fusion algorithm to improve the overall prediction accuracy. In order to evaluate the advantage of the fusion algorithm against other fusion techniques, we proposed a vector of *n* values for each classifier prediction. Each position in the vector is associated with one of the concepts and the position value represents the classifier prediction score of that concept. Because a classifier (e.g. CNN) can predict one label for each region, the prediction score is saved to the label position at the vector and the rest of values of other labels are set to zeros. If there are two classifiers, then there will be a vector of *2*n* values. The results of using SVM and KNN as prediction models for that fusion technique are shown in Fig. 7.13. The results show that the algorithm based technique outperforms the SVM and KNN fusion technique.

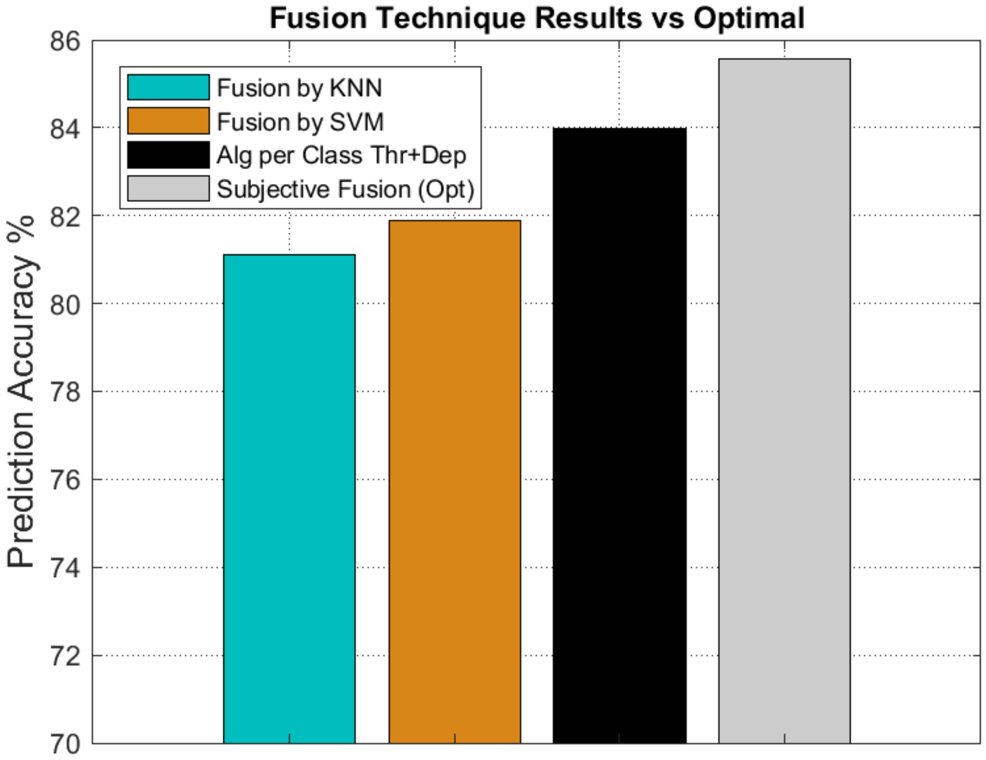


Figure 7.14 Fusion algorithm versus SVM, KNN and subjective optimal fusion

1.3 Rejection Principle

In this section we would like to incorporate sample rejections for samples that falls in the marginal rejoin between two or more classes. The rejection of these samples would add confidence to our system since they cannot be classified in either class.

Rejection as a proposed option in the prediction system increases the system accuracy by rejecting ambiguous image region that none of the classifiers can predict properly. In our dataset, multiple regions are noisy and hard to be precisely predicted; therefore the need to reject these regions is highly recommended. However, this task is associated with the challenge of how to propose such an option and increase its accuracy. In this essence, we propose a rejection model that is trained on rejecting erroneously labeled image regions. We hypothesize that our previous analysis of the dependency metric and other scores can be used to discriminate between included and rejected regions. Our experimental results provide support of this hypothesis.

In order to propose a rejection model, there is a need to determine the model main attributes that are used to decide whether the predicted label of an image region should be accepted or rejected. Based on algorithm 7.1 and its results, the per_class threshold approach with dependency metric shows the best performance as is shown in Fig. 7.12. If we analyze the fusion algorithm workflow, we will find that the algorithm decides on the labels with four consequence conditional statements or simply criteria. Three out of these four criteria considers the prediction score of the predicted label and whether this score is above the threshold value or not. In case of the scores of the predicted label for both classifiers are below their threshold values (line 16), then the decision of the fusion algorithm only depends on this case of the metric in use (e.g. dependency metric) with uncertainty. Therefore, the rejection model is mainly proposed to help with this scenario, and trained to review all the predicted labels resulted from the last stage of the fusion algorithm to decide whether to accept or reject any of them. The focus on the fusion algorithm last stage (line 16-19) is due to the fact that, the majority of fusion algorithm wrong labels are resulted from this stage. For example, the fusion of FV-HOG and CNN-FT has 21 wrong labels where 20 of these wrong labels are resulted from the algorithm last stage.

In order to propose a working rejection model, several attributes have been evaluated for that purpose. The following attributes are considered for each classifier of the fusion algorithm:

1- The predicted label dependency score.

2- The prediction score value of the predicted label.

3- The optimal threshold value of the predicted class.

4- The precision value.

5- The normalization value.

The normalization value is the position of the predicted score relative to the classifier highest and lowest prediction score as shown in Fig. 7.8. Therefore, there is a total number of ten attributes that are used in predicting the rejected labels. The performance of the selected attributes is shown in Fig. 7.15. The figure shows that selecting all of the mentioned attributes achieves the highest rejection accuracy.

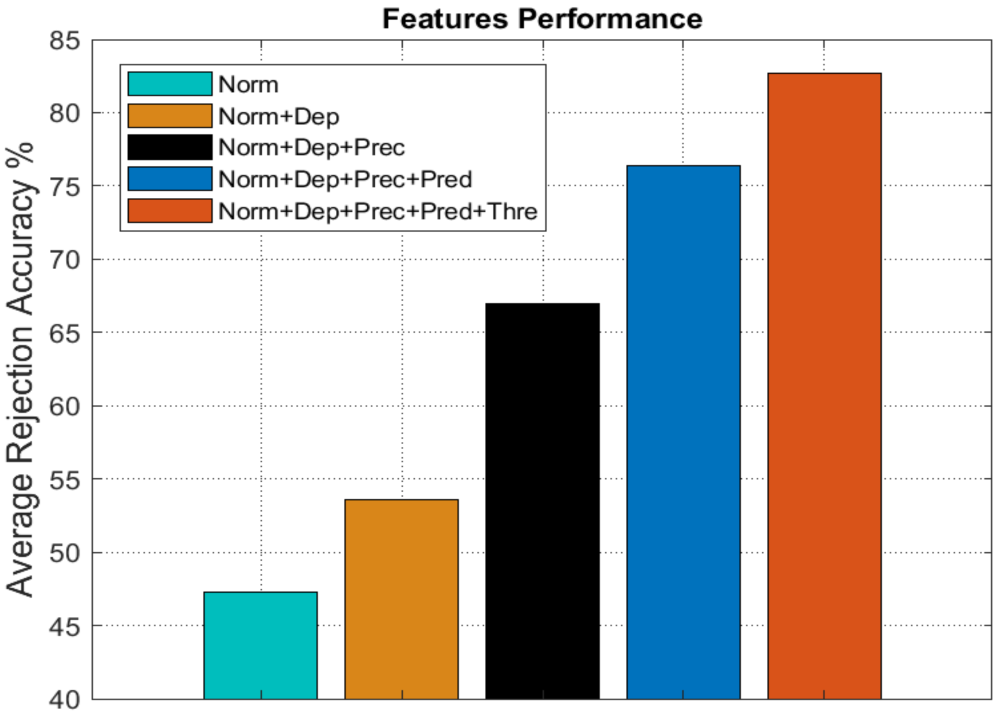


Figure 7.15 Performances of the selected attributes of the rejection model

In Fig. 7.15, Training the rejection model is based on two testing datasets representing the labels of the algorithm last stages as we mentioned before. After training the fusion algorithm for predicting the fusion labels, a second phase training is performed for the rejection model based on the results of the previous phase. Fig. 7.15 shows that the selected attributes that are used to train the last stage of the fusion algorithm on the rejection principle contribute positively to the accuracy of the rejection model. The selected attributes trained by using the Discriminant Analysis (DA) as a classification technique to discriminate between the acceptable and rejected labels. The DA classifier achieved the highest accepted/rejection accuracy compares to state of the art classification techniques as shown in Fig. 7.16.

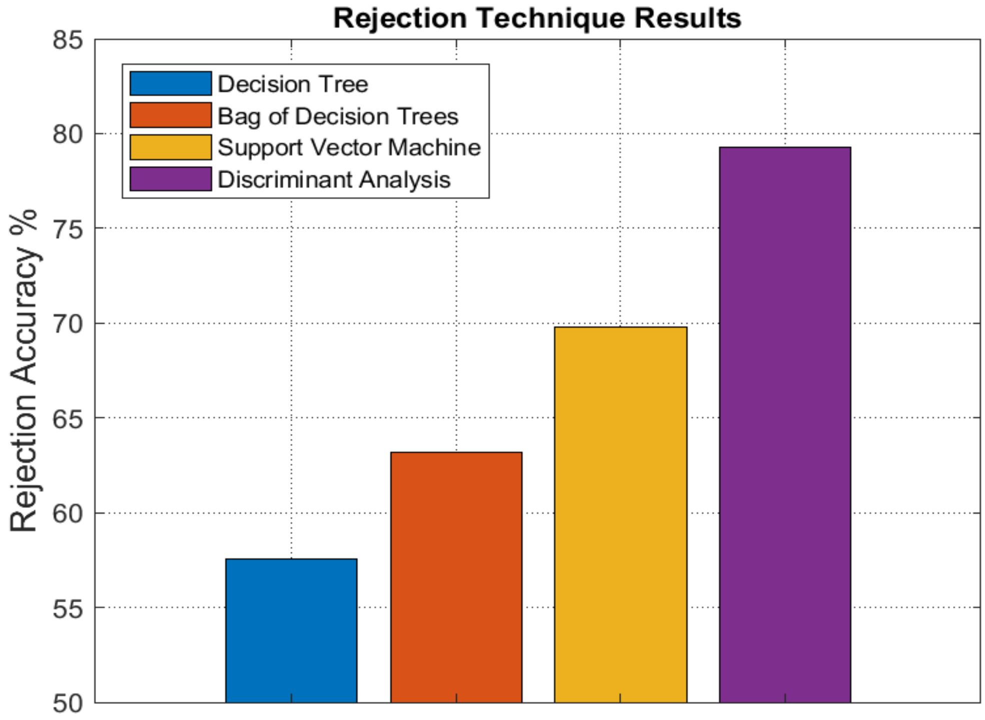


Figure 7.16 Performances of several classification techniques with the rejection model

In Fig. 7.16, for 106 predicted labels with uncertainty, the accept/reject model decides correctly on 84 labels. Therefore, the total number of errors is 22 such that 9 errors are false negative (reject correct output), and 13 errors are false positive (accept wrong output). The accuracy of the rejection model is 79.2 %. Samples of the results are shown in Fig. 7.17-19.

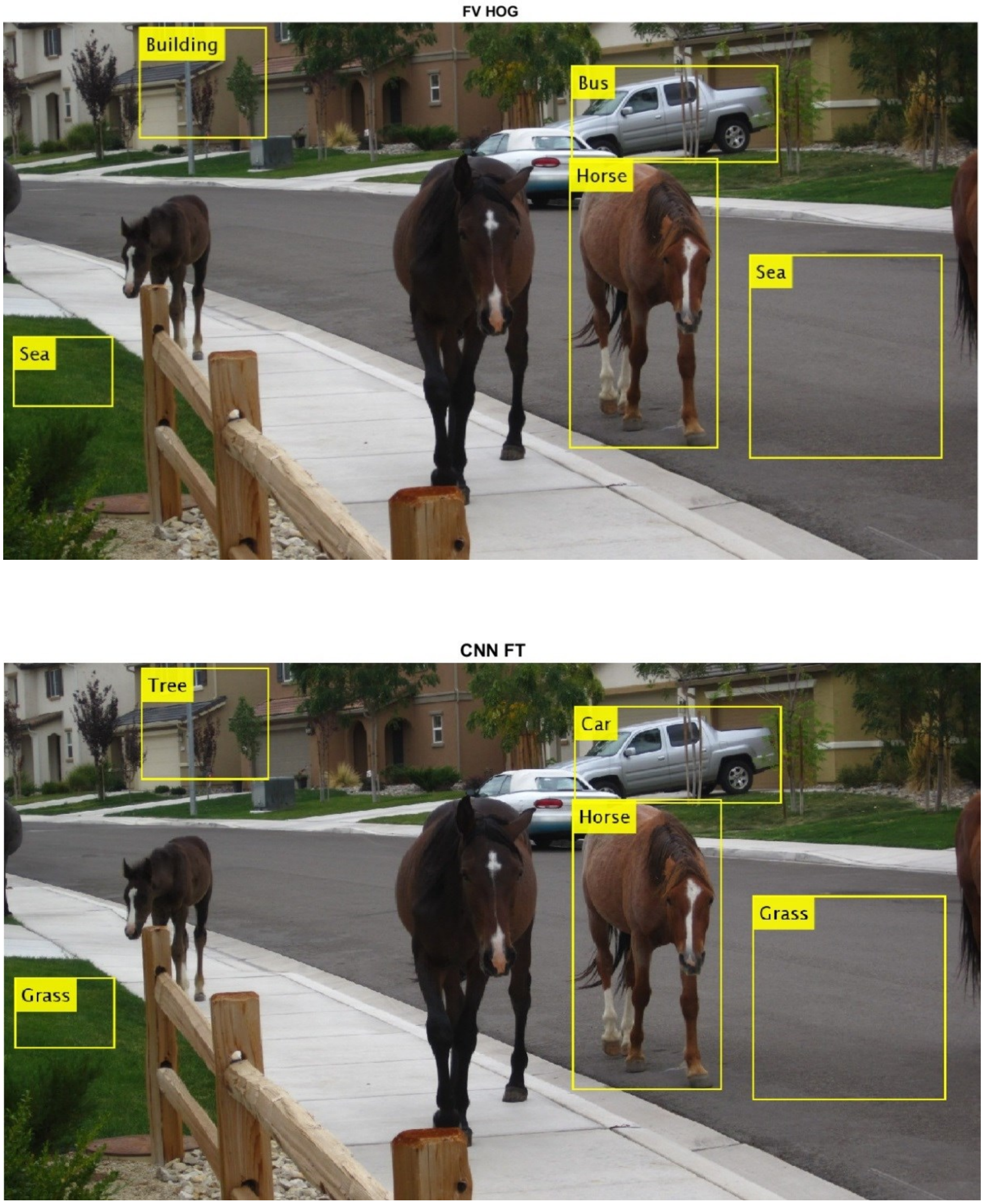
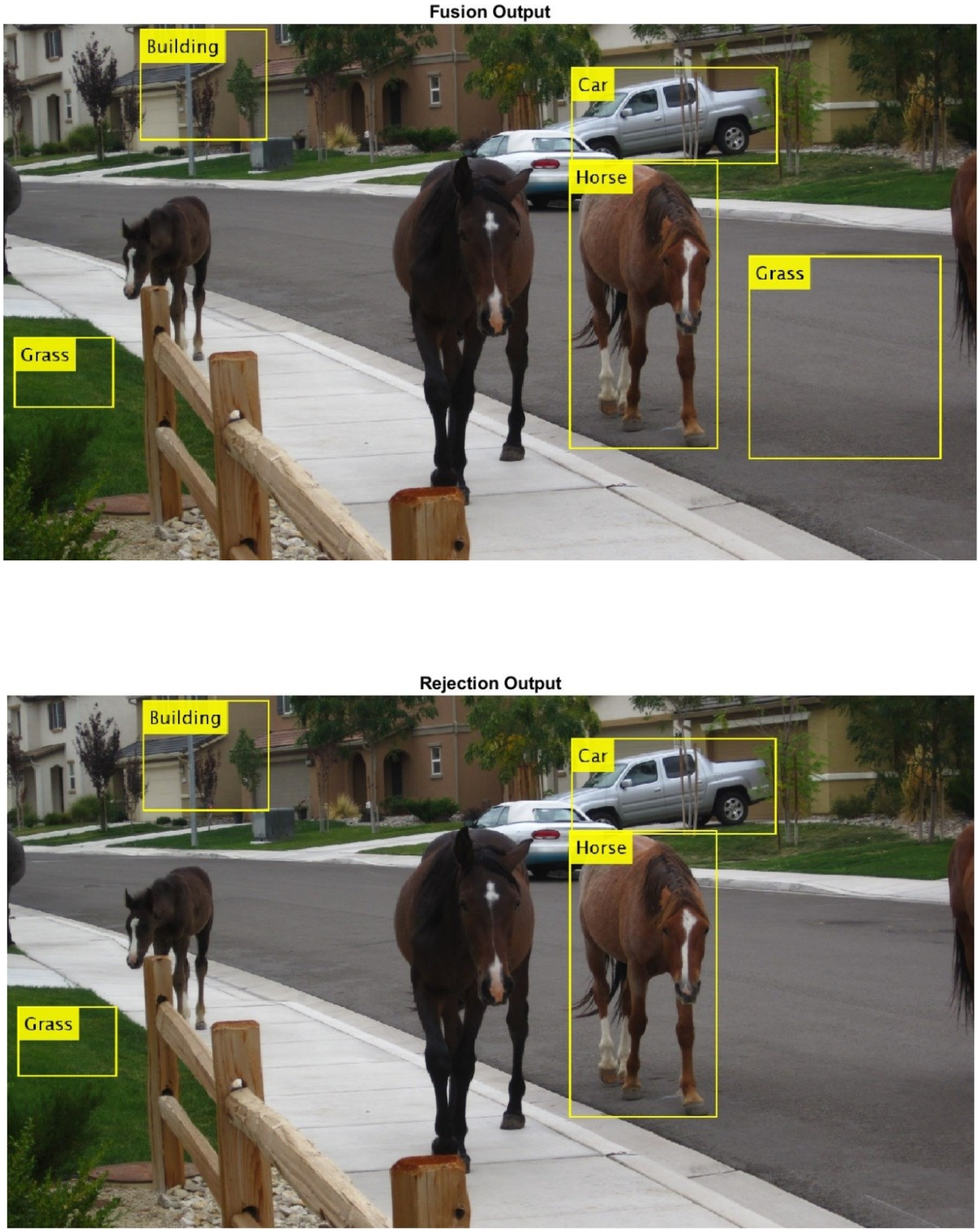


Figure 7.17 Outputs of the standalone classifiers (Source: Connie [56])



Fusion Output

Rejection Output

Figure 7.18 Outputs of fusion algorithm and the rejection model (Source: Connie [56])

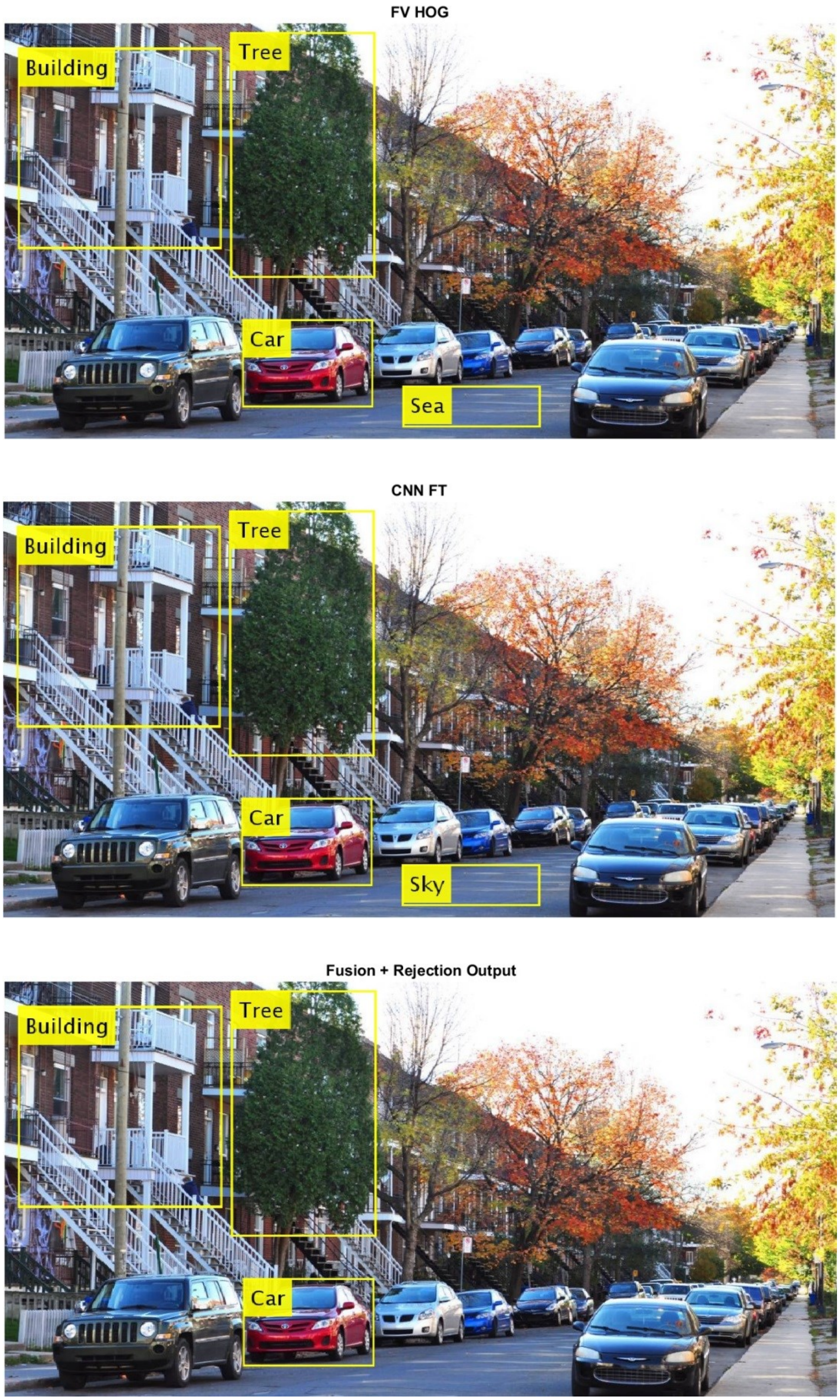


Figure 7.19 Outputs of classifiers, fusion and rejection model

7.4 Summary

In this chapter, we proposed a modified version of the fusion algorithm that improves the performance of the current classification methods. The algorithm is evaluated using a new constructed dataset for multi-label images. Different fusion techniques are proposed with different decision criteria. The results indicate the advantage of our fusion approach in improving the overall prediction precision. Refining the resulted labels by excluding wrong labeled regions is also considered by incorporating a new rejection model.

8 CONCLUSIONS, IMPLICATIONS AND FUTURE WORK

8.1 Conclusions

Automatic annotation have come into the spotlight due to their semantic importance and broad applications; Multimedia databases, human-machine interaction systems, intelligent transportation systems, and multimedia wireless sensor networks are few of many other domains that can directly benefit from the advances in automatic annotation. However, the current research work in this area does not consider the effect of the mutual relations between the various components of the annotation process.

In this dissertation, we present a comprehensive study of the effect of each component in the overall performance of the annotation system. To improve the overall performance of the annotation system we introduce a novel multi-layered annotation architecture that utilizes feedbacks between its main components as an enhancement tool. The proposed model has many advantages when annotating image regions. First of all, our model not only takes into account the correlation between produced labels, but it also can select the label with the highest accuracy to predict other regions in the recognition layer. Eventually this can lead to a ranking of possible labels of each region according to their context. Because traditional annotation approaches only focus on the similarity between unlabeled images and labeled ones, they often do not consider the untagged image context which is a very important factor that is unique for each image. Our proposed tagging system can also be extensible which enables an external source to provide extra keywords or features that can enrich the original tag. Extensible tagging is another important feature that can increase the accuracy of the annotation system and make it progressively dynamic.

The evaluation task gives us insights into the main building block needed to design the training stage for single-label and multi-label systems. It also highlights the importance of the fusion techniques in increasing the accuracy of the system. Different fusion techniques are proposed with different decision criteria. The results indicate the advantage of our fusion approach in improving the overall prediction precision. Our tagging model maximizes the utilization of the ground truth information. The model refined its output results by incorporating a new concept of rejection.

8.2 Practical Implications

The implications of our techniques emphasize their applicability in image databases, in general, as well as other important datasets such medical images, and similar multimedia data.

For multimedia data, our techniques can be extended to include voice and sound data. Sound signals can be handled using voice recognition techniques that works on a very similar set of feature vectors in their databases. Therefore, the developed algorithms and techniques in this dissertation can be implemented to enhance the recognition phase for multimedia data as well.

The practical implication of our techniques in the medical field is also verified by our initial results with medical image databases. As a proof of concept, two groups of medical images, namely the bloody images and the ink marked images are collected from G*lioblastoma* GBM [57]. The modified version of the fusion algorithm that combines the strength of deep learning and the power of color histogram image descriptors is evaluated and shows improvement in the classification results. These initial results clearly indicate that our techniques can be used to enhance the classification accuracy of medical datasets.

For image database design, the implications of our study lie in the following:

- Ease of access
- Low storage requirements
- No need to transmit every image to destinations rather replacing it with the tags associated with a link to the original image location.
- Tags can also contain metadata that includes copyright (e.g. time stamps of uploading) as well as owner original descriptions.

Therefore, the implications of our study are not limited only to the annotation pipeline, and they can be extended to many other related systems.

8.3 Future Work

In this study, it has been shown that our model and techniques can enhance classification of images as well as increase annotation accuracies. However, some other possibilities for more improvement in the annotation process can be achieved if we use the resulted tags to predict new associated tags and discover new regions. In addition fine tuning our model to deal with larger databases is another area that can be explored in future work.

Training our model to deal with partial or occluded objects is also an interesting domain to explore. Using larger databases and different types of images such as medical images are currently under consideration as a future research work.

REFERENCES

[1] K.Hirata, T.Kato, "Query by visual example -Content- based image retrieval," Third international conference on extending Database Technology, 1992J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon.

[2] R.Chakarvarti, X. Meng," A study of color histogram based image retrieval," Sixeth International conference on Information Technology.IEEE,2009.

[3] J. Huang, S. Kuamr, M. Mitra, W.-J. Zhu, R. Zabih, "Image indexing using colour correlogram," in: Proceedings of the CVPR97, 1997, pp. 762–765.

[4] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, P. Yanker," Query by image and video content: the QBIC system," IEEE Computer 28 (9) (Septem- ber, 1995) 23–32.

[5] R B.B.Chaudhuri,N.Sarkar,"Texture segmentation using fractal dimension," IEEE PAMI17(1)(1995)72–77..C.

[6] Gonzalez, R.E. Woods, "Digital Image Processing," third ed., Prentice-Hall, 2007.

[7] K.-L. Lee, L.-H. Chen, "An efficient computation method for the texture browsing descriptor of MPEG-7," Image and Vision Computing 23 (2005) 479–489.

[8] S.B. Park, J.W. Lee, S.K. Kim," Content-based image classification using a neural network," Pattern Recognition Letters 25 (2004) 287–300.

[9] Z. Lu, S. Li, H. Burkhardt, "A content-based image retrieval scheme in jpeg compressed domain, " International Journal of Innovative Computing, Infor- mation and Control 2 (4) (2006) 831–839.

[10] D. Zhang, A. Wong, M. Indrawan, G. Lu, "Content-based image retrieval using Gabor texture features," in Proceedings of the First IEEE Pacific- Rim Conference on Multimedia, Sydney, Australia, December 2000, pp. 392–395.

[11] T.Zahn, Roskies, Ralph Z., "Fourier descriptors for plane closed curves" IEEE transaction on computers, 1972

[12] M.Teague," Image analysis via the general theory of moments" journal of the optical society of America, 1980

[13] D. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints", *Int'l J. Computer Vision*, vol. 2, no. 60, pp. 91-110, 2004.

[14] AlexNet : https://github.com/BVLC/caffe/tree/master/models/bvlc_alexnet

[15] Krizhevsky, A., I. Sutskever, and G. E. Hinton. "ImageNet Classification with Deep Convolutional Neural Networks. " *Advances in Neural Information Processing Systems*. Vol 25, 2012.

[16] ImageNet dataset : http://www.image-net.org/

[17] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, "Imagenet: A large-scale hierarchical image database", *CVPR*, 2009.

[18] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng, “Nus-wide: A real-world web image database from National University of Singapore,” in *Proc. CIVR*, 2009.

[19] J. Corso, A. Yuille, and Z. Tu, ”Graph-shits: Natural image labeling by dynamic hierarchical computing” In Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition, 2008.

[20] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, “ Learning hierarchical features for scene labeling” IEEE Trans. PAMI, 35:1915-1929, 2013.

[21] F.D. Frate, F. Pacifici, G. Schiavon, C. Solimini, “Use of neural networks for automatic classification from high-resolution images,” IEEE Transactions on Geoscience and Remote Sensing 45 (4) (April 2007) 800–809.

[22] X. Qi, Y. Han, “Incorporating multiple SVMs for automatic image annotation,” Pattern Recognition 40 (2) (2007) 728–741.

[23] Zhi-Hong Deng , Hongliang Yu, Yunlun Yang “Image Tagging via Cross-Modal Semantic Mapping”. Proceedings of the 23rd ACM international conference on Multimedia, 2015

[24] Cusano, Claudio, Clocca, Gianluigi, and Schettini, Raimondo “Image annotation using svm”, in electronic imaging 2004, pp.330- 338 . International society for optics and photonics.

[25] D. Grangier and S. Bengio, " A discriminative kernel-based approach to rank images from text queries", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 8, pp. 1371-1384, 2008

[26] T. Ojala, M. Pietikainen, T. Maenpaa,," Multi-resolution gray scale and rotation invariant texture classification with local binary patterns," IEEE transaction on pattern analysis and machine intelligence,2002.

[27] G.-H. Liu, Z.-Y. Li, L. Zhang, and Y. Xu, "Image retrieval based on micro-structure descriptor," Elsevier, Pattern Recognition., vol. 44, no. 9,pp. 2123–2133, 2011.

[28] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in Computer Vision and Pattern Recognition, vol. 2. IEEE, 2006, pp. 2169-2178.

[29] J. Pujari and P.S. Hiremath, "Content based image retrieval using color,texture and shape features," Proceedings of the International Conference on Advanced Computing and Communications, pp. 780-784, 2007.

[30] M. Pietikainen, T. Ahonen, and V. Takala, " Block-based methods for image retrieval using local binary patterns," Proceesings of the 14th Scandinavian Conference on Image Analysis, pp. 882–891, 2005.

[31] Corel 1K dataset: http://wang.ist.psu.edu/docs/related/

[32] X. Yuan, J. Yu, Z. Qin and T.Wan, "A SIFT-LBP retrieval model based on bag-of features" In ICIP, pp. 1061-1064, 2011.

[33] Caltech 101 dataset : http://www.vision.caltech.edu/Image_Datasets/Caltech101.

[34] F. Perronnin, C. Dance, "Fisher kernels on visual vocabularies for image categorization", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'07), pp. 1-8, June 2007.

[35] Florent Perronnin , Jorge Sánchez , Thomas Mensink, "Improving the fisher kernel for large-scale image classification," Proceedings of the 11th European conference on Computer vision: Part IV, September 05-11, 2010, Heraklion, Crete, Greece

[36] Dengsheng Zhang , Md. Monirul Islam , Guojun Lu, "A review on automatic image annotation techniques," Pattern Recognition, v.45 n.1, p.346-362, January, 2012

[37] F.D. Frate, F. Pacifici, G. Schiavon, C. Solimini, "Use of neural networks for automatic classification from high-resolution images," IEEE Transactions on Geoscience and Remote Sensing 45 (4) (April 2007) 800–809.

[38] K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman, “The devil is in the details: an evaluation of recent feature encoding methods,” in Proc. BMVC., 2011.

[39] G. Carneiro, A.B. Chan, P.J. Moreno, N. Vasconcelos,“Supervised learning of semantic classes for image annotation and retrieval,” IEEE PAMI 29 (3) (2007) 394–410.

[40] Xiangdong Zhou , Mei Wang , Qi Zhang , Junqi Zhang , Baile Shi, “Automatic image annotation by an iterative approach: incorporating keyword correlations and region matching,” Proceedings of the 6th ACM international conference on Image and video retrieval, p.25-32, July 09-11, 2007, Amsterdam, The Netherlands

[41] Sanghoon Lee, Lee, Y. Zhao, **M. Masoud**, M. Valero, S. Kul, and S. Belkasim. "Domain specific information retrieval and text mining in medical document." In Proceedings of the 6th ACM Conference on Bioinformatics, Computational Biology and Health Informatics, pp. 67-76. ACM, 2015.

[42] Sanghoon Lee, **M. Masoud**, J. Balaji, S. Belkasim, and R. Sumderraman, “A Survey of Tag-based Information Retrieval”, International Journal of Multimedia Information Retrieval, springer , 2016.

[43] Pascal VOC datasets : http://host.robots.ox.ac.uk/pascal/VOC/

[44] Corel 5k dataset : http://www.ci.gxnu.edu.cn/cbir/Corel/

[45] **Mohamed Masoud**, Sanghoon Lee, and Saeid Belkasim, “Statistical Based Image Tagging”, IEEE/ACM/WI International Conference on Web Intelligence, Omaha Nebraska, 2016.

[46] **Mohamed Masoud**, Krishanu Sarker, Saeid Belkasim, Iman Chahine "Automatically generated semantic tags of art images", In Proc. of IEEE International Conference on Signal and Image Processing Applications (ICSIPA). 2017, pp. 155-158

[47] **Mohamed Masoud**, Saeid Belkasim, and Iman Chahine “Deep Learning Fusion algorithm for Arts Categorization”, IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE), Ontario, 2017.

[48] Raj, Alex Noel Joseph, and Vijayalakshmi GV Mahesh, "Zernike-Moments-Based Shape Descriptors for Pattern Recognition and Classification Applications." Advanced Image Processing Techniques and Applications. IGI Global, pp. 90-120, 2017

[49] Bay, H., A. Ess, T. Tuytelaars, and L. Van Gool, "SURF:Speeded Up Robust Features." Computer Vision and Image Understanding (CVIU).Vol. 110, No. 3, pp. 346–359, 2008

[50] J. Matas, O. Chum, M. Urban, and T. Pajdla "Robust wide baseline stereo from maximally stable extremal regions," In Proc. BMVC, 2002.

[51] Dalal, N. and B. Triggs. "Histograms of Oriented Gradients for Human Detection", IEEE Conference on Computer Vision and Pattern Recognition, Vol. 1, pp. 886–893, 2005

[52] A. Krizhevsky, I. Sutskever, and G. E. Hinton. "Imagenet classification with deep CNN." In Proc. NIPS, 2012.

[53] Csurka, G., C. R. Dance, L. Fan, J. Willamowski, and C. Bray. "Visual Categorization with Bags of Keypoints. " Workshop on Statistical Learning in Computer Vision. 2004

[54] A.Vedaldi and K.Lenc,"MatConvNet-Convolutional Neural Networks for MATLAB" Proc. of the ACM Int. Conf. on Multimedia, 2015.

[55] A. Vedaldi and B. Fulkerson. "VLFeat - An open and portable library of computer vision algorithms." In Proc. ACM Int. Conf. on Multimedia, 2010.

[56] Thatcher, C. "walking”, Online image. Flickr. Sep 06, 2014. https://www.flicker.com/photos/67616811@N00/15579393672/

[57] Medical dataset: Cancer.digitalslidearchive.net