A COPULA BASED METHOD FOR FISH SPECIES CLASSIFICATION

By

Raj Singh Dhawal

B.Eng. Computer Science
Agra University, India, 2007

THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE
IN
COMPUTER SCIENCE

UNIVERSITY OF NORTHERN BRITISH COLUMBIA

June 2015

ABSTRACT

The purpose of this thesis is to develop a method for classification of the species of a fish given in an image. This method uses the state of the art multi-dimensional image descriptor HOG (Histogram of Oriented Gradients), and colour histograms to create representative feature vectors. In this work copula theory have been used to summarize the multi-dimensional features. Copula theory has been used extensively for analysing the bivariate data; however, not much exploration has been done to find its application for analysing multivariate data. This work is one of the few attempts where copulas have been used to analyse multivariate data. The classification accuracy of this method is comparable with other reported methods.

Table of Contents

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENT

I would like to express my gratitude to my supervisor Dr. *Liang Chen* for being an exceptional mentor. I want to thank him to encourage me for both the research and personal fronts.

I am very thankful to my committee members Dr. *David Casperson* and Dr. *Pranesh Kumar* to enrich me with their brilliant analysis and excellent comments and suggestions that greatly improved my manuscript. I am also very grateful to University of Northern British Columbia to provide me the golden platform/opportunity to research on my favourite topic. I want to dedicate my research thesis to all.

Moreover, I want to thank to all my friends who have been of strong support for my academics and social engagements.

And a special thanks to my Parents, my Sister and my Wife for their love, prayers and support.

CHAPTER 1 INTRODUCTION

The objective of this work is to develop a novel approach for the classification of fish species. This work is a subordinate level classification problem; subordinate level classifications are currently drawing a lot of attention from research community [1, 2, 3, 4, 5, 6, 7]. The following subsections discuss respectively the problem, the proposed approach and the contributions made through this work.

1.1 Problem Description

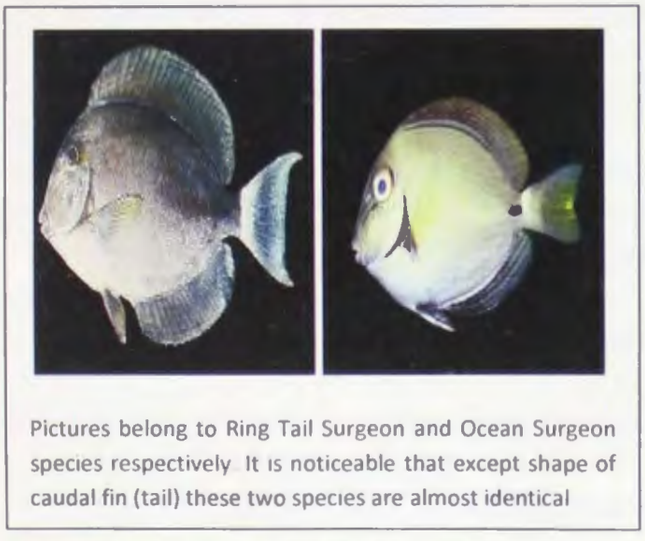
In simple words, we can write the objective for this work as follows:

'To develop an algorithm for classification of a fish in the given image'

In nature there are different species for the same class of organism; for example, there are numerous species in the class 'Fish'. Species are grouped under a 'Family' which are based on similar characteristics. Species from different families may significantly differ from each other whereas the species from same family can have many similarities; as a result, sometimes it is difficult to discriminate among them. Figure 1 shows two fish; belonging to two different species (same family); they seem to be identical apart from a slight difference in the shape of their caudal fins (tails).

The classification of fish species is a challenging task because the images of same fish species can show significant differences in the fish's attributes when taken in different conditions. For example; the colours of a fish in an underwater picture can look completely different when compared to a picture taken in the open environment. In Figure 2 there are two images of 'White Spotted Surgeon Fish' (Refer to Images 1and 2). These pictures are taken in an open environment, and under water conditions respectively. A large colour difference can easily be noticed in the two pictures.

Additionally, we can observe a huge difference in the attributes in the images of same species, even when the photos are taken in same environment. The same fact can easily be observed in Images 3 and 4 in Figure 2, both of the images are of 'Doctor Surgeon Fish', and are taken in underwater conditions; however, the difference in the attributes of the fish is huge. Usually such kinds of diversity are not observed in other species' datasets such as the 'Bird datasets of Caltech' [14]. The classification can be more complicated when images have different background as highlighted in the images of Figure 2.



Pictures belong to Ring Tail Surgeon and Ocean Surgeon species respectively. It is noticeable that except shape of caudal fin (tail) these two species are almost identical

Figure 1: Images of lookalike species

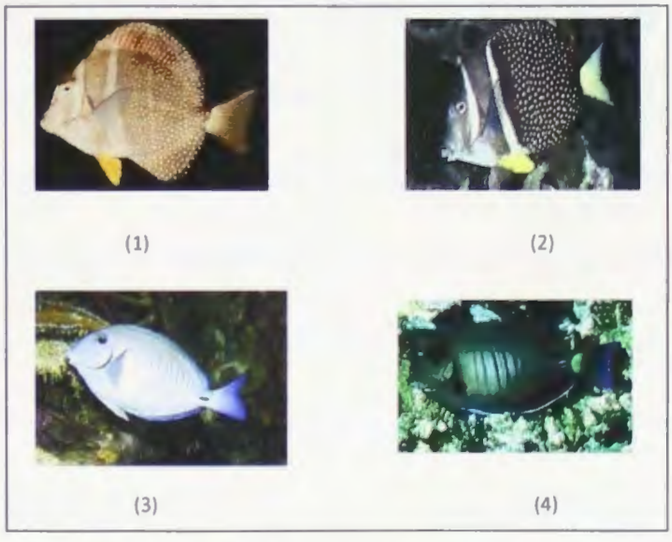


Figure 2: Images in different environment

It is quite clear, that for classification of fish images, we need to catch subtle differences amongst the subjects.

The problems of subordinate level categorization is distinguishing among similar kinds of objects, such as computer tables and dining tables, rather than a basic level categorization, such as distinguishing between a table and a chair. This type of standard categorization is defined as '*Fine-grained-Categorization*' [1, 2, 3, 4, 5, 6, 7, 8]. Generic object classification includes the daunting task of differentiating amongst objects that only have subtle differences [9].

Traditional classification approaches do not work well with Fine-grained-Categorization problems. A possible reason traditional classification approaches do not work is because of the way the features are encoded within them. For example, in the visual vocabulary based approach, the image patches are encoded using clustering which results in the loss of fine details, which are important for Subordinate level classification [1, 8, 9, 10, 11].

Some recent work in the area of *Fine-grained-Categorization* deals with the classification of different datasets such as flowers [12], larvae [13], bird species [1, 13], leaf nodes of image net [15] *et cetera*. Many different approaches have been tried for Fine-grained-Categorization. These approaches include encoding the local image patches to visual vocabulary [8, 9, 10], using sparse coding to include more information in visual vocabulary [12, 16], and approaches that use pre-defined information about the images using annotations on object parts in order to extract subtle information. The annotation based approaches have shown promising results [2, 4, 7, 17]. A growing number of researchers are using the latter approach because of its ability to provide significant object articulation and subtle distinctions.

1.2 Proposed Approach

Through our exploration of the research in this area, we have not found any well known work in the area of image-based classifications of fish species.

The fish species classification method presented here is inspired by the annotations based approaches. These approaches have the proven ability to catch the subtle distinction amongst the objects [2, 4, 7, 16]. For species classification, a comparison based on parts is intuitive since the differences among the categories are very

small, and it is difficult to acknowledge these differences at a global level. In fact, for subordinate level problems, objects will have the same parts which will make comparisons easier. For example, the caudal fin, which is an essential part of fish, should be compared with respective caudal fins of all other fish species in order to have a finer level comparison. To achieve this, an obvious choice for comparison, would be to select an 'image patch' that includes the caudal fins of the classes of fish and compare them. Following this concept, our approach is based on a 'part based comparison' for analyzing the local patches of images, of different body parts, in order to catch the subtle differences among the subjects.

To analyse the subject 'Fish', we have designed our approach to analyse the local patches based on the specific body parts of the subject fish. The analysis, based on body parts, is fruitful in two major ways. Firstly, by providing a swift and smooth way of local comparison; as a result, we can make a comparison of the features, extracted in the reference of a specific body part. For example, test features extracted around the eyes are compared with training features extracted around the eyes only. In this way, our approach can catch the finer details of the subject. Secondly, the influence of the background information is restricted.

We extract two basic features from each local patch. The first feature extracts shape information and the second feature collects information about the colours. We select both of them (the features) because they perform an important role in identifying any object even in everyday life. We use both of the features in concatenation to construct collective information while making the comparisons. Our method uses Copula theory to identify the relationships among the multiple dimensions of our features. Our feature

vector for each local part is a concatenation of the outcome of fitting copula onto the multi-dimensional features and the colour histogram. Thus, our feature vector comprises information about the shape and the colour. We have used a simple linear distance method to compare the test and the training vectors.

For our experiments, we created a database of images downloaded from the Fish Base which is one of the largest relational databases of information about fish; this database contains practically all the information about all fish species known to the world to date [19]. We pre-processed the selected images for our experiments in order to create part location labels.

At this stage, there is no other work found on the classification of the fish images; hence, the results of this work cannot be compared directly with any other proposed work. However, to support our method, we tried experiments with the 'Bag of Words approach' on our database and compared the results. We also did experiments with CALTECH database [14] and the obtained results are competitive with other proposed work [1, 13]. Furthermore, it shows that our method is extendable on other databases. A detailed discussion and analysis of the results are provided in later chapters.

1.3 Major Contribution

Our method is simple yet it gives very good results on both the fish database and the standard Caltech Bird database [14]. In our algorithm, we have used very basic image attributes as gradient vectors, colour histogram, linear classification *et cetera*. These attributes were selected because they are well known and easy to understand. We used the copula theory to analyse our multivariate feature space by measuring the

dependency among the different dimensions. It is one of the few attempts where efforts have been made to harness the potential of copulas in multivariate image analysis.

For our experiments, we have created the database of fish images with proper annotations of their body parts. The fish images, when taken in different environments, show a lot of diversity among them; hence, making the classification task more complex. We believe that this database, if enhanced further, could provide a very good base for testing the subordinate classification methods.

CHAPTER 2 LITERATURE SURVEY

Species classification is a problem of subordinate level category. A natural approach for this type of classification focuses on *'characteristics defined on part level'*. A possible reason for this may be the same physical structure of the objects (presence of same parts in each object). In the basic level categorization, the presence or absence of any part is determinative, where as in subordinate classification distinction has to be found at the part level.

Cognitive psychology theory states that subordinate classification is determined based on the differing properties of the respective parts in the objects [32]. Subordinate categorization depends on identifying how 'part properties' vary across different categories for the respective parts [4, 32]. This requirement to identify the subtle differences on the part level often poses the challenge even for highly knowledgeable humans, and thus posses a great challenge in computer vision task.

The key for any approach for subordinate level categorization resides in encoding of features. Traditional approaches often do not perform well with subordinate level categorization [4]. In the traditional approach of object classification, the image patches are encoded to form a dictionary of visual words using the clustering method, which results in the loss of finer details, which are important for fine-grained classification [1].

Researchers in the recent past have tried different approaches for subordinate level categorization that can broadly be categorized into Codebook based approaches, Annotations based approaches and Template based approaches. A brief survey of recent work done in all these approaches is provided in trailing subsections:

2.1 Codebook-Based Approaches

Codebook based approaches has been trendy in generic image classification systems [9, 10, 11, 12]. Codebook based approach encodes the local image patches to code words, often referred as visual vocabulary, for coarse coding of the images.

Gabreilla *et al.* [9] use the vector quantization of affine invariant descriptors of image patches for generic categorization. They use the Harris affine detector [32] for finding the image descriptors based on affine region. The selections of affine regions are based on the iterative selection of the position, scale and the elliptical neighbourhood of the points. This approach of selecting the image patches doesn't specifically have any defined description such as eyes, wings, tyre *et cetera*. This encoding is thus a coarse encoding which results in the loss of finer details, which plays an important role in subordinate classification [1], and is one of the reason why the bag of words approach does not work well with *fine-grained* categorization [1, 4].

In [8, 10, 11, 12, 17], new methods have been suggested to improve the performance of the codebook-based approaches. Svetlana *et al.* [10] propose the use of feature extraction at finer sub regions of the images using the spatial pyramid. They come up with a geometrical invariant 'orderless bag of feature' image representation at finer resolutions over fixed sub regions, to bring more efficiency into the classification. All feature vectors are clustered into M discrete types under the assumption, that only the same type feature can be matched with each other. This study focuses on subordinate recognition and doesn't involve global object recognition. The objective was to capture the context of the image to provide more informed search for the specific objects. As they have mentioned in their work '*if the image, based on its global description, is likely to be*

a highway, we have a high probability of finding a car, but not a toaster'. This approach focuses on some finer details for the image context.

The improvement in the performance of the spatial pyramid matching is achieved by Jinjun *et al* [11]. They replace the vector quantization with a locality constrained linear coding. This scheme projects each descriptor to a local co-ordinate system, and the projected coordinates are integrated by pooling, to generate the final representation. This scheme with linear classifier performs remarkably better than the traditional non-linear spatial pyramid system.

This method has a basic assumption that the locality is essential with sparse coding [12, 34], and achieve reduced computational complexity of the spatial pyramid system by making them work with the linear classifiers. This work uses a simple representation of the image context and showed promising results for the generic image classification.

Zho *et al.* [16] propose a nonlinear coding method for the image patches called super vector coding. It formulate the use of piecewise constant approximation for the vector quantization coding, in order to achieve a lower function approximation error. This coding improves the performance of the bag of visual word approach. All of the above works have focused on basic level image classifications only, and try to improve the information coding to cover whole image context.

Aditya, Yao and Li Fei Fei [7] explore the use of code book based approaches for the *fine-grained* image categorization. They focus on finding the discriminative image patches, to calculate fine image statistics for categorization. They use the concept of randomization and discrimination respectively in order to handle feature space and model

the information about images. To locate the image patches that contain the point of interest, the authors propose the use of rectangular image patches of arbitrary width and height at an arbitrary location. These arbitrary rectangular image patches generate a very dense sampling space. In order to explore this dense sampling space authors use two concepts: *Discriminative training* to extract the information in the image patches and Randomization to explore the dense sampling space efficiently. They create a random forest with a discriminative decision tree algorithm. Every node in the tree is a discriminative classifier and is trained on a pair of image patches.

Yao *et al.* [1] extract only the fine image statistics useful for *Fine-Grained-Categorization* by creating a dense sampling space, selecting only those patches of the images that are relevant for the given classification objective. A simple example of this dense space sample can be understood from the task where we need to identify whether in a given picture, a person is playing a guitar, or just holding it in hand. For this objective, we want image patches below the human face that are related to the guitar playing activity. To create the sample space, the random forest framework is used in which each node a classifier is trained on one or a pair of image patches. The random forest framework allows analysing a set of image patches, and selects the best image patch in each node. In this way, the redundancy in the sample is reduced greatly. Further, using discriminative classifiers enable a stronger decision tree with small correlation [7]. Once the sample space is created, the approach for extracting statistics follow the codebook approach where for each image patch, SIFT features are extracted followed by the use of k means clustering to create vocabulary code words.

It is natural that for subordinate level categorization, we should focus only on small portion of the image that is relevant for classification objective. Considering the whole image context does not serve our purpose as it brings in more information that is irrelevant, and makes the classification task more complex. The other main approaches for subordinate level classification involve annotations based approaches and template based approaches that focus only on some relevant part of the image for the classification.

2.2 Template-Based Approaches

Template based approaches are gaining popularity with some prominent work recently being done in area of basic level classification and scene classification tasks. These approaches primarily use the high level image representations to generate image response features such as object detectors [35, 37], action recognition of human body [36] *et cetera*.

Li, Su, Xing and Fei Fei [35] argue that high level feature space such as direct object of the images, may offer a better visual recognizer for handling of random natural images. They create a high level representation of the natural images based on object sensing, built on a generic collection of labelled objects. They use the response of two state of art detectors, SVM object detectors [37] for identifying the objects like table, humans and, 3D pop up models [38] to detect material objects like sky, roads, to form an image representation.

This image representation is called the Object Bank because images are represented as collection of objects. For their work they select a few hundred of objects from the images, available in the dataset for the purpose of training. The numbers of detectors are narrowed down by selecting the intersection set of most frequent objects.

Selected object detectors are then trained using the bounding box information to create the object bank.

Finally, the method uses an LR classifier on the high dimensional representation, which exploits object and sparsity (feature sparsity vial_1, object sparsity via l_1/l_2 and joint sparsity via $l_1/(l_1 + l_2)$) hand in hand. This method shows very promising results with the basic level object classification and the scene classification. However this method is based on the outcome of detectors that are able to classify among different shapes *i.e.* basic level classification, which may not suit well for the subordinate level classification.

Maji, Bourdev and Malik [36] use the 3-D orientation of the head and torso to create a distributed representation of 'pose and appearance' to identify persons. They include various information in their formulation such as 'interaction among objects and other people' in the image to improve the action detection. Their work is an extension of the poselets [17, 40] that use the annotation data of joint location of people, to train detectors and SVM classifiers to detect body parts.

They estimate a 3-D pose of the head and the torso using the bounding box and create a poselet activation vector which has an entry for each poslet type, reflecting the degree to which poselet type is active in the person. To define the poselet type, they use bounding box of a person to check which poselet type is consistent with it. These rotations are used to calculate estimated rotation around Y to decide on the poselets. This method focuses on creating a high dimensional human pose space for action recognition and pose estimation.

Torresani, Szummer and Fitzgibbon [37] introduce a new image descriptor: Classemes. Classemes are output of a large number of weakly trained object category classifier and have a good accuracy on the object recognition. The system works on satisfaction of three criteria: *novel categories* (represent new category as a set of training images, to train classifier to avoid distance computation between database image and the novel training images), *compact descriptors* (suggesting 2 orders compact descriptors) and *simple classifiers* (because they can be run efficiently on large datasets).

Their system is a classifier combination that uses the output of predefined classifiers. Basically this method selects one category from the available categories and gathers a predefined number of training images by issuing a 'query on the selected category'. Then for any given image a classeme vector is defined as the concatenation of outputs of one vs. all classifier for each selected category. A feature selection is done on the given classeme vectors of the training images with reduced dimensionality. Final denotation is encapsulation of the output of classeme learning.

Now to test any new given category, the training images are used to do the classeme learning for the new category. Generated classeme vectors for the new category can be used with classifier taking the classeme vectors as input. Again, a high-level image representation is used in this work for object recognition. The authors argue that weakly trained classeme might not carry good semantic information, and say that one can see classeme as highly nonlinear random projections [37].

Yao, Bradski and Fei Fei [1] have used a template based approach for *fine-grained-Categorization*. They have implemented a high level representation of the images using templates. They propose a direct template matching process using a large

number of randomly generated templates to overcome the basic problem of losing subtle information that is critical for subordinate categorization. For classification, they built a classifier based on bagging algorithm using the aggregation of discriminative classifiers.

They generate a large number of rectangular templates from the training images and, then represent an image by response scores of matching itself with the templates. An image is represented by different features such as colour, gradient *et cetera,* and a template is represented as collection of location and feature pairs in the space. For matching, the image is rescaled and then matched with each template. The measure of similarity is based on the algorithm proposed in [43]. A max pooling on spatial pyramid is used to transform the response map, generated from the template matching process, to produce a feature vector.

Fei-Fei *et-al* [1] argue that normal classifiers like the single SVMs may not be a good option for classification because the template generation is random; hence, many of the templates may be from the uninformative regions. To overcome this problem they proposed the aggregation of outputs of the set of classifiers with the condition that the correlations between the classifiers are small. They focus on full usage of all the templates matching information rather than reducing the dimensions [1]. They claim to achieve one of the best results available till date with the specified datasets.

Thus we see that most of the template-based approaches focus on formulating a high-level image representation. It is also clear from the survey of the two approaches that for a subordinate classification we need to target the subtle differences among the categories as done by the major research works done in this area [1, 8].

2.3 Annotation-Based Approaches

To achieve better articulations and find subtle distinctions among the images, annotation based approaches are used by various researchers [3, 43, 18, 44]. These approaches seek human inputs in reference to the images such as asking for annotations on the body parts. Sometimes these approaches even keep humans in the loop [43, 44] to click on certain objects for the processing of algorithm. The main advantage of annotation based approaches is that they allow for better part based attribute formation to capture the subtle differences and keep out unusable information for the classification. A major disadvantage of these approaches is that putting up annotations is a tedious and costly task and may require some domain experts for finding the object key points [1]. Wah, Branson [43] and Steve,Catherine [44] are some of the recent contributors to this area.

Steve *et al.* [44] have presented a hybrid method for object classification. Their method focuses on classification of such objects that need expertise to get identified such as species classification. Their method is based on asking interactive questions regarding the visual content of the image for classification from the user.

For any given image, at each step the algorithm exploits the image content, history of questions and their responses to select the next question. For any given answer, the user is asked to select a confidence value which is used to calculate a prior term. For selecting the set of questions they use the criteria of *'maximum information gained from set of questions and the set of responses for each of them till last step'*. The information gain is basically computed as the *'probability of information gain of posing an additional question'* and is used in decision trees.

To implement the framework on images they use a classifier trained on offline data using the algorithm proposed by Andrea Vedaldi's [45]. This classifier works on multiple kernel based learning which combines geometric blur and SIFT features using a spatial pyramid and, one vs. all SVMs. The classifier is used on each image to estimate the probabilistic output and is continuously updated by gathering more answers for further questions.

This work uses the human recognition abilities along with the strength of computer vision. The authors argue that object recognition is in niche stage and it may take long for development of proper algorithm that can give good results by themselves. However, it is clearly evident that this hybrid approach focuses on specific part based feature extraction for achieving good results, supporting the basic assumption of subordinate classification.

Another annotation based approach for multiclass recognition was proposed by Wah, Branson and others [43]. The method involves the user input where it asked users to click on object parts and answer binary question. Just like the last work, algorithm is designed in such a way that it has the ability to select the most informative questions for the user to identify the object class. Each class of the homogenous category is represented using a unique, deterministic vector of attributes.

Attributes are associated with parts, whose ground locations are already provided in terms of coordinates, scales and aspects. The work focuses on estimation of a distribution over the location of each part, taking into consideration the image content, the history of questions and their responses using the probabilistic models. Proposed work calculates the attribute probabilities on the set of *'predicted part locations'* and,

probabilities for *'part locations'*. Attribute probabilities are calculated in terms of the sigmoid parameter using a sigmoid function on the classification scores for each attribute from the binary classifier. Part detection probabilities are calculated based on detection scores that are modeled as a sum over unary and pair-wise potential log. This calculation of the detection score is inspired by the work of Felzenszwalb *et al.* [46]. Parts and aspects are defined semantically, and are handled using the mixture model and the weight parameter for appearances, and spatial terms are learned jointly using a structured SVM.

For questions selection the authors focus on using the informativeness of user responses taking into account the expected level of human error, information, annotation time and spatial relationship between different parts using the framework described in [44]. The method evaluates every possible answer to the question and re-computes the class probabilities. Authors in their work have made assumption of linear relationship between bits of information and reduction in human time. They use this assumption to minimize the expected amount of human time spent by encoding the expected number of bit of information for any feature. It gives the criteria that *'best question to be asked is the one with largest ratio of information gain against expected time to answer'* [43]. This method focuses on using the human expertise in extracting features for the parts of given object. This method performs well with challenging datasets for subordinate classification.

Farrell *et al.* [3] propose a subordinate level classification method by using pose normalized appearance using a volumetric poselet scheme. They use a pose classifier to classify the part appearance and shape information. They associate image pattern

parameters with location, scale and orientation to define a mapping from pixels to pose normalized space for fine-grained categorization.

For their formulation they describe the object as constellation of volumetric parts and, focus on utilizing the shape and arrangement properties for the categorization. The pose normalization provides a surface parameterization on the volumetric parts that is used for representation of the non-parametric appearance. The method creates a parameterized space of surface-normal of the patch descriptors for the comparisons. The poselet approach used in this formulation is inspired by the work [17, 47].

While implementing the model, they select two ellipsoids for the head and the body respectively. Then the pose parameters are used to determine the transformation for mapping unit sphere points to the ellipsoid's surface. This transformation is completely reversible giving back the image points to the unit sphere. The normal is computed for all the points that have been transformed to the ellipse surface. These normal are used to extract the tangent patch on the tangent plane centered at the point. The patch is again projected back into the image and we extract the features (such as SIFT) from that. They yield a pose normalized appearance by concatenating the normal vector and extracted descriptor. This process is followed on all visible ellipsoidal part to form non parametric representation.

They represent the objects as volumetric primitive templates as it allows estimating geometric attributes such as part location, size, aspect, orientation *et cetera*. These templates allow encoding the intrinsic characteristics of the object and capturing the essence of the shape. The classification is performed using the stacked evidence tree models [48].

In this method they use a 2 level categorization. For any image, features are passed through the random forest and all label distribution vectors are collected to form an evidence vector. Then a multiclass classifier is applied to the evidence vector to give the final category. This categorization allow not only mapping of the visible portion into space but also effectively telling which parts in the space be used for classification.

2.4 Aggregator Models and Methods

In any of the image processing applications such as face recognition, indexing, image classification *et cetera.*, researchers design their approaches using image descriptors such as Histogram of Oriented Gradients(HOG), Scale Invariant Feature Transform (SIFT) and other descriptors to detect and describe image features. Usually the entire image descriptors are defined in a multi-dimensional space, hence aggregation models such as Bag of words [9], Fisher vectors [51] *et cetera* are used to summarize the information to provide a representative vector for an image.

Multivariate analysis is the core of machine vision algorithms. Most of the state of the art methods for multivariate analysis are based on the theory of probabilistic graphical models that allow marginal and posterior computations, estimation and model selection [49]. The dependency measure is fundamental to almost all known machine learning algorithms such as feature selection, clustering, structure learning *et cetera.*

There have been numerous methods introduced for the aggregation of image descriptors, and among them Bag of Visual Words is probably the most widely used in the area of image descriptor analysis. In the Bag of Words approach, multidimensional image descriptors are extracted at the image key points [49, 50] and then clustered them into a visual codebook. This codebook is then used to map any image into a

representative vector which is basically the approximation of the multivariate probability distribution functions of the image descriptors.

In the same pattern, the fisher vector also approximates the distribution of the image LIDs using the Fisher kernels. The fisher method finds the similarity between the probability distribution function of the image descriptors and the global distribution function of all the image descriptors. In short we can easily say that these approaches involve the approximation of joint probability of the of the image descriptors directly or indirectly.

Copula theory [29, 30] states that marginals play an important role in the multivariate theory. The copulas fundamentally can be seen as a measure of dependency. It has been very efficient for modelling the real value distributions; however, the use has been focused only for few variables, mostly bivariate, whereas on the other hand the probabilistic graphical model have been extensively used for high dimensional domains but their capacity in real valued scenarios is limited [49]. The potential of copulas in the field of probabilistic graphical model has not been explored much, even with the fact that graphical models have limitations in the context of real valued measurements [49, 52]. Copulas offer a flexible way of studying probability distributions and they are closely related to the dependency identification which forms the backbone of the machine learning [52]. In essence we can see a lot of potential of using the copulas for analysis of high dimensional domains. In some recent machine vision works, copulas have been used as image quality measures [28], modelling of image key points [27], image registration [53, 54], codebook design [50], dual polarization synthetic aperture radar image analysis [51] *et cetera*. Copulas along with concepts of machine learning can offer a huge

potential in developing new techniques for dependency measurement in high dimensional space. Copulas have found extensive use in area of quantitative finance and other engineering subjects such reliability analysis, turbulent combustion, Medicine, weather research, *et cetera*. However, the use have been majorly restricted to bivariate cases.

Redi and Merialdo [27] use copulas to provide an image descriptor based feature. Authors use copula theory to do a probabilistic analysis of multiple dimensional image descriptors. They use a mono-variate modelling approach [52] to calculate the marginal distribution of the image descriptors and then create a joint distribution of the descriptors using a Gaussian copula.

Their copula provides the information about multiple dimensions of the image descriptors based on the dependencies among marginal distributions. The authors use the joint distribution obtained from the copulas for the image classification. The image feature proposed in this method directly stores the parameter of the joint distribution; hence provide a more informative feature in comparison of global codebook and fisher vectors, which approximate the joint distribution through vector quantization, and the parameter adaption for GMM fitting respectively. This method uses the aggregation of image descriptors based on the copula theory to form an image representation that is more discriminative than the bag of words approach and the fisher vectors for image encoding.

Zahir and Kashanchi [28] propose the use of copulas for the image quality measurement. Their results with the copulas are comparable with the results of other state of the art methods. Their method is based on evaluating the mutual information from images using the Gaussian copula. They propose a full reference image quality algorithm by using the Gaussian copula for calculating the information content of the reference and

distorted images, to be used as a quality level representation. The authors apply the copulas on the wavelet coefficients selecting the sub band 4 steerable pyramids for both the reference and target image. The authors used the copula methods to calculate the marginal densities to extract the mutual information among the images and, use it to order the images according the level of noise.

There exist a large number of families of copulas and each of them performs better with specific type of data [29]. Gaussian copulas have been prominently used in the area of computer vision [27, 28, 50, 51, 53, 54].

CHAPTER 3 KEY ALGORITHMS USED

The proposed classification method uses state of the art algorithms for feature extraction, representation and classification. The proposed work states that, after pre-processing the images, we compute the Histogram of Oriented Gradients (HOG) features and colour histograms on the defined point of interests. The extracted HOG features are then summarized in a feature vector using the copula theory and then the resultant vector is concatenated with the colour histogram to form the final feature vector. Finally, the classification is done using the linear prediction algorithm. The trailing sub-sections will briefly cover each of these algorithms.

3.1 Histogram of Oriented Gradients

The HOG feature, were first introduced by Navneet Dalal and Bill Triggs [23] to detect humans in images [23, 24]; later these features were also used for detecting objects [25, 26].

The HOG features work on localized portions of images, and count the occurrences of gradient orientations in these localized portions. The HOG features also focus on evaluating the well normalized local histograms in a dense grid where the basic idea is to finely characterize the shape of the local object by the distribution of the edge directions [24]. The HOG descriptor is the combination of the histograms of gradient directions of the spatial regions; hence, it is different because it describes the object through a global feature rather than a collection of local features. The following block diagram shown in Figure 3 displays the feature extraction process.

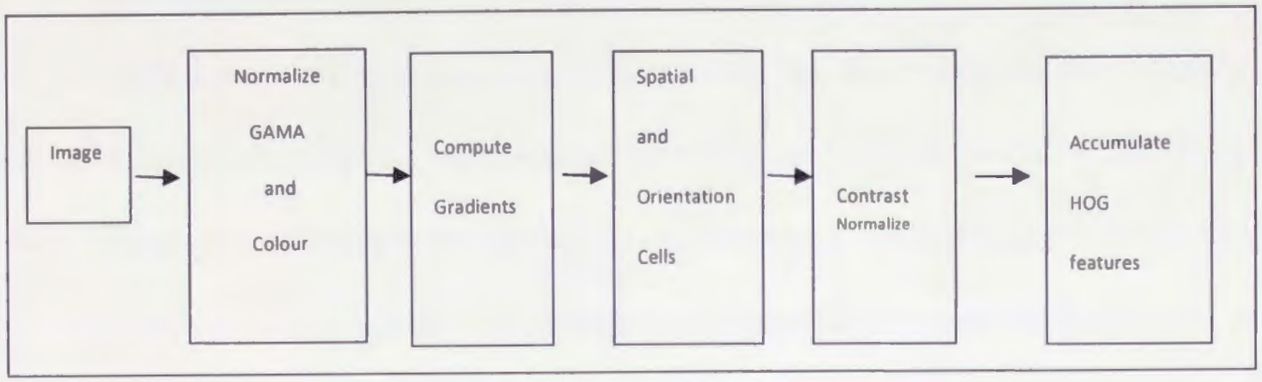


Figure 3: Calculation of HOG features

The HOG features have numerous advantages, for example, the local shape captured by the features are invariant to any local geometric and photometric transformations [24]. Any translation, or rotation, makes little difference in the feature vector if they are much smaller than the local bin size.

There are two main variants of the HOG feature. The first one was proposed by Dalal and Triggs in their original work [24], and the other one is the enhanced version of the HOG features that is proposed in [55]. In this work, we use the enhanced HOG feature. In both algorithms, the main framework remains same, except the fact that the enhanced version uses a boosting framework based on an enhanced variable size and utilizes a fixed Gaussian template to interpolate the weight of each pixel. In short, the enhanced version uses directed and undirected gradients [24, 55]. Dalal and Triggs worked only with undirected gradients and did no compression in the results (they represented their feature in 36 dimensions). The enhanced version does the compression of the results, and represents the feature in a 31 dimensional space.

A brief outline of the major steps for calculating the HOG descriptor is given in the following sub-sections.

3.1.1 Gamma and Colour Normalization

This step is optional in the HOG feature computation since it has a modest effect on the performance of the algorithm. The reason for this might be the subsequent normalization of the descriptors in further steps that achieve the same results. Gamma normalization is used to code and decode the luminance in order to maximize the use of the bits in identification of light and colour. In the algorithm, colour information is used whenever available; however, when the inputs are restricted to gray scale, the performance is marginally reduced [23]. The gamma compression of the colour channels provides a better performance than log compressions (log compressions are too strong) [23]. As mentioned earlier, this step is optional, since it does not have noticeable effect on the performance of the algorithm.

3.1.2 Gradient Computation

The 'Gradient vector' is one of the basic and most prominent concepts used in the image processing for edge detection. It measures the change in the value of each pixel in both the X and Y directions. Gradient vector can be computed for each pixel in the image. The gradient vector is computed by putting two values together: The first value is obtained by subtracting the left value from the right value (showing the rate of change in X direction) and, the second value is obtained by subtracting the upper value from the lower value of the target pixel. By putting these values together, we can get the gradient vector for the pixel. Using these values, we can calculate the magnitude and the angle of the vector. These gradient vectors can be shown as an arrow to the edge in the pixel. Gradient vectors have high performance in edge detection and feature extraction.

The methods through which the gradients are computed play a very important role in the performance of the detectors. The gradients are computed by applying *1-D* centered derivative masks in both the horizontal and vertical directions. In their work, Dalal and Triggs test several discrete masks for gradient computation; these masks include 1-D point derivatives ([1, -1] un-centred, [-1, 0, 1] centred and [1, -8, 0, 8, -1] cubic corrected), 3×3 Sobel masks and 2×2 diagonal ones. They claim that they get the best results using the simple *1-D* mask [-1, 0, -1]. When larger masks are used, performance always decreased. The results with Gaussian smoothing are poorer than the results of the simple *1-D* mask.

3.1.3 Orientation Binning

Orientation binning is basically the creation of the cell histograms. Each cell is a local spatial region for which we have orientation bins. These bins are used to accumulate the weighted votes for each pixel for an edge orientation histogram channel, based on the orientation of the gradient element centered on it. Cells have either a rectangle or a radial shape and are evenly spaced, either unsigned ($spaced\ over\ 0^0\ to\ 180^0$) or signed($spaced\ over\ 0^0 to\ 360^0$). The votes can be classified by the magnitude itself, its square, its square root, or by a part of the magnitude based on the soft presence/absence of an edge on the pixel. Dalal and Triggs [23] find that the classification based on magnitude provides the best results. It is found that increasing the number of bins results in better performance; however, the best results are achieved with 9 bins and any further increase only has a subtle effect.

3.1.4 Descriptor Blocks and Normalization

In this case, contrast normalization is necessary in order to have high performance; this is because the gradients are spread over a wide area that may have variation in illumination and background contrast. In general, a basic requirement of all available normalization schemes is to group cells into larger connected spatial blocks. After grouping cells into blocks, each block is then normalized separately. The final descriptor is defined by the vector of the components of these normalized cells over the detection window. For normalization, the overlapping of blocks is allowed since it provides better performance. A possible reason for this is that the overlapping allows multiple contributions of components from a scalar cell to the final descriptor. Dalal and Triggs propose two types of blocks for the grouping of cells, rectangular and circular. These groupings are partitioned into rectangular spatial cells and log polar style respectively. The rectangular block has three components based on the partitions of the block: cells per block, pixels per cell, and channels per cell histogram. The number of each component within the block may vary depending on its size. In their study, Dalal and Triggs found that a 3×3 cell block, a 6×6 pixel cell, and 9 channels are the optimal parameters for a rectangular block. In the circular block, there are two variants: the first is with a singular circular central cell and the other is with a central cell divided using angular sectors. There is a stack of gradient weighted orientation cells within each spatial cell. It is observed, that with fewer bins, the descriptors achieve better results. The circular layout is defined by four parts: The number of angular bins, the number of radial bins, the radius of the central bin, and the expansion factor for radii [23]. The optimum performance is achieved when using two radial and four angular bins. It was observed

that further increasing the number of radial bins did not show any improvement in performance; whereas any increase in the number of angular bins resulted in poorer performance.

Dalal and Triggs also try to apply the Gaussian weighing in order to improve their results. The Gaussian spatial window is applied within each block in order to weigh the pixels around the edge. It provides a small improvement in results when using rectangular blocks; whereas in the case of circular blocks, Gaussian weighing provides no benefits.

For normalization, four different methods are tried. For each block, if v is the non-normalized descriptor vector, $\| v \|_k$ is its k norm (where k=1, 2) and ϵ is a small constant. Then under the four schemes, the normalization factor will be one of the following:

$$\textbf{L}_2\textbf{ norm}: f = \frac{v}{\sqrt{(\|v\|_2^2 + \epsilon^2)}} \quad (1)$$

L_2 hys: L_2 norm followed by clipping limiting the max values of v to .2

$$\textbf{L}_1\textbf{ norm}: f = \frac{v}{\|v\|_1 + \epsilon} \quad (2)$$

$$\textbf{L}_1\textbf{sqrt}: f = \sqrt{\left(\frac{v}{\|v\|_1 + \epsilon}\right)} \quad (3)$$

In the experiments, it is found that normalization resulted in much higher performance. Of the four schemes, L_1 norm provide slightly poorer results when compared to the other three, which provided similar results.

The following Figure shows the computed HOG features for sample images from our target database:

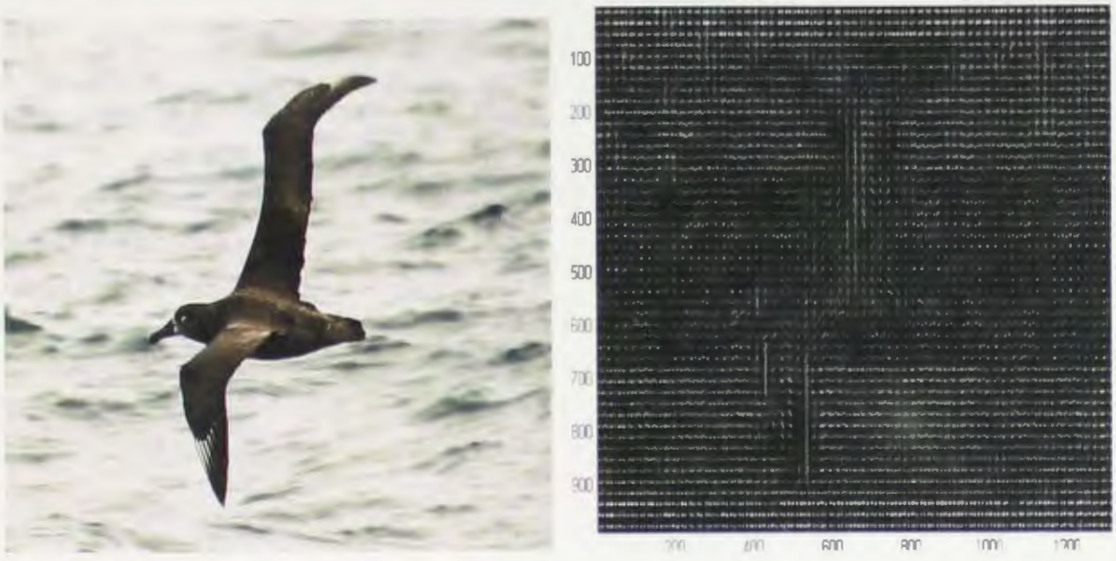


Figure 4:HOG features for a sample picture from CALTECH database

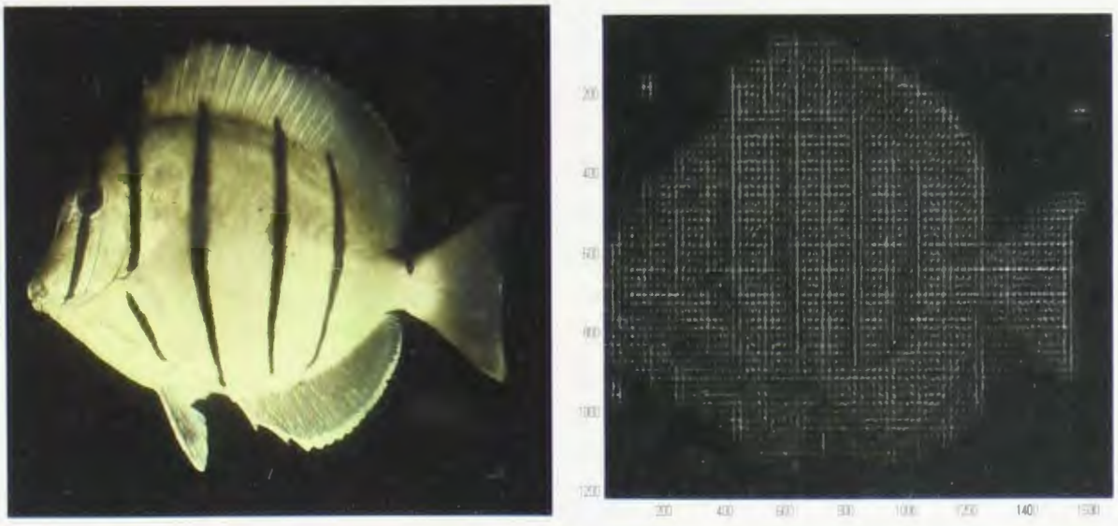


Figure 5:HOG features for a sample picture from our FISH database

Figure 4 and Figure 5 show the HOG feature computed for two different subjects: a bird and a fish. Our research problem is a subordinate-level classification where we need to classify the class of the same breed of object, rather than classifying the class from a set of multiple breeds of object. To understand this more precisely, the research problem is to classify the species of the fish amongst all available fish classes, rather than classifying the fish against other objects like man, lion, *et cetera*. As a result, we cannot generalize the features on global level; hence, it is required to look at more precise parts.

We follow the same approach and extract the HOG feature from the targeted cropped partition of the image. This procedure is shown in detail in following chapter.

3.2 Copula Theory

Most of the state of the art feature extraction algorithms that are used today are defined in multi dimensional space. The biggest challenge when using the extracted feature is dealing with the multi- dimensionality. Usually these features are summarized in a feature vector using different statistic, or other summarizing features. In some works, dimension reduction is also performed before creating the feature vector. In our work, we used the copula theory for summarizing the multi dimensional HOG feature. Copulas are used to model the dependencies among multiple dimensions of the features; these dependencies form the basis of the feature vector.

Copula theory represents the distribution of the image descriptors through its marginal and a copula function. A copula is a multivariate probability distribution function that has uniform marginal distribution. It makes it easy to model the joint distribution of random variables through their marginal distributions and a copula function. In simple words, *'Copulas are functions that join multivariate distribution function to their one dimensional marginal distribution function'* [30]. Copulas are used to get the joint distributions of the variables when their marginal distributions are known. For given k variables, with their respective k marginal distributions, there exists a copula function C that relates these k marginal functions together in order to get their joint distribution function [30, 31]. Copula theory can be used when analyzing multivariate problems.

The probability distribution function for any vector in a k dimensional space can be decomposed into k marginal distributions and an associated copula function. Marginal distributions give information about the probability of each of k variables in the k dimensional space, and the copula function defines the dependencies among the marginal distributions. This mapping is based on marginal values and does not involve computationally expensive multidimensional searches [28]. Because of this property, copulas are one of the most widely used tools for multivariate analysis in financial and medical data. We are focusing on the use of copulas for analysing the marginal distribution of multidimensional local image descriptors of the image.

Here a brief description of the formulation and conditions of the copulas is provided. First the definition of copula is given which is followed by the formulation showing how the copulas link marginal distribution with joint distribution. The formulation and the definition have been defined for two dimensions, for the ease of understanding. The concepts can easily be extended to higher dimensions. As mentioned earlier copulas are distribution functions, with uniform marginal distribution, that enables the calculation of the joint distributions when the marginal distributions are known [28, 30, 31]. Copula basically joins multiple distributions to their one dimensional joint distribution; this fact is shown in Figure 6.

Let there be two random variables X and Y in a two dimensional space with their cumulative distribution functions given by $F_X(x)$ and $G_Y(y)$ respectively. Then a copula function C can link the two marginal of X and Y on their joint distribution $J_{XY}(x, y)$ as shown in Figure 6.

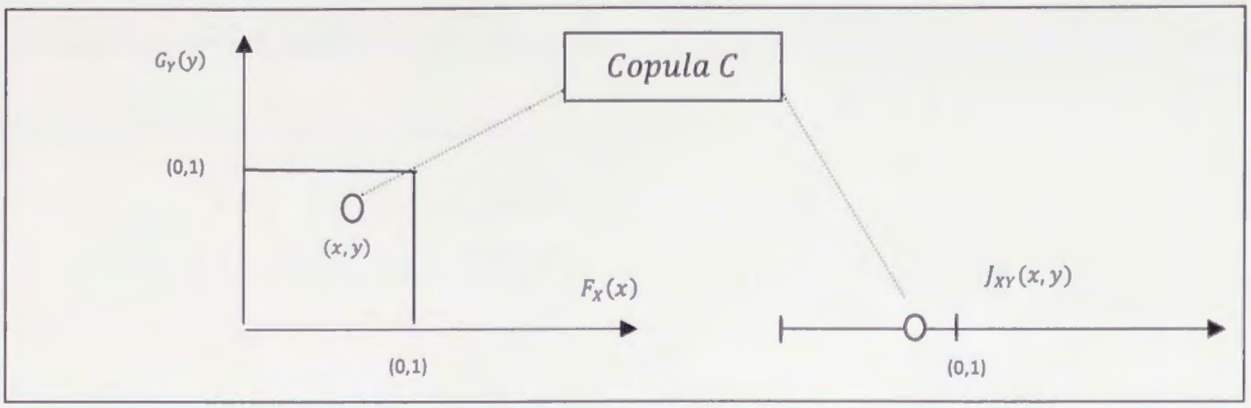


Figure 6:2 D Copula(C) corresponding a point from unit square on a joint distribution

A two dimensional copula function (C) must have the following properties:

i.) $Dom\ C = S_1 * S_2$ where S_1&S_2 are subset of I where I= [0, 1] any power of I refers to one dimension of space

ii.) C is grounded, *i.e.*

for every $u\ and\ v$ in I

$$C(u,0) = 0 = C(0,v) \quad \& C(u,1) = u \,\&\, C(1,v) = v \tag{4}$$

iii.) C is 2-increasing, *i.e.*, for every u_1, u_2, v_1, v_2 in I and if $u_1 \leq u_2$ & $v_1 \leq v_2$ then

$$C(u_2,v_2) - C(u_1,v_2) - C(u_2,v_1) + C(u_1,v_1) \geq 0 \tag{5}$$

Let's say that in a two-dimensional space there is random vector $V_{XY} = \{x, y\}$ and the marginal distribution cumulative function of each dimension is $U = F_x(x) = [P(x \leq X)]$ and $V = F_y(y) = [P(y \leq Y)]$ respectively. Joint cumulative distribution of vector is

given by $F(x,y) = P[x \leq X, y \leq Y]$. Now according to Sklar's theorem [31] a copula C can be defined as a unique mapping between the ordered pair of marginal values of vector V_{XY} and its joint distribution function:

$$F(x,y) = C\left(F_x(x), F_y(y)\right) = C(U,V) \tag{6}$$

Following the above equation we can write:

$$C(U,V) = F(x,y) = F(F_x^{-1}(U), F_y^{-1}(V)) \tag{7}$$

subject to F_x and F_y being continuous. From this equation it is obvious that we can create a copula given the distribution information of the multivariate variables. The copula itself describes the multidimensional vector giving information about the dependency in the structure and their mapping to a multivariate Cumulative Distribution Function. These dependencies, by themselves, provide a sufficient base to compare different classes. However, as mentioned earlier, we can obtain a joint probability density function to represent the vector from the copula information. To find the joint probability density function $f(x,y) = P[x = X, y = Y]$ we need to compute the density of copula or the derivative of the cumulative distribution. From Equation (7), we obtain joint density function:

$$C(u,v) = \frac{\partial^2 C(U,V)}{\partial U \partial V} = \frac{f(F^{-1}(U), F^{-1}(V))}{f(F^{-1}(U),\ f^{-1}(V)} \tag{8}$$

Using a copula we can define the dependencies among the components of the multivariate variables without having detailed knowledge of the function that describes the marginal.

There are several Copula families that have been introduced, the most commonly used are: Gaussian Copula, Student *t* copula, Archimedean Copulas (Clayton, Gumbel, Frank, Joe *et cetera*). Every family does not fit well with every application or field. Some of the copulas may work well with image data while some of the others may not. It depends on the structure of the copulas. For processing image data, *'Gaussian Copula'* has been preferred by researchers because of its simple calculation [27, 52]. We also decided to proceed with the Gaussian copula for our application. In the following section we have provided a brief discussion about the Gaussian copula and why we feel that it may go well with our objective.

A Gaussian copula follows an elliptical distribution over a unit cube $[0,1]^d$and its structure has a multivariate normal distribution. A Gaussian copula C_R^G , where G refers to the Gaussian and R refers to the parameter matrix of correlations, for a multidimensional vector V is defined as:

$$C_R^G(V) = \alpha_R\left(\alpha^{-1}(V_1), \ldots\ldots, \alpha^{-1}(V_d)\right) \quad \textbf{(9)}$$

where α^{-1} is the inverse of univariate normal cumulative distribution function and α_R is joint cumulative distribution function of a multidimensional vector V with mean zero and correlation matrix R defined over the space of $[0,1]^d$. The joint density function associated with copulas in (9) is expressed as:

$$D_R^C(V) = \frac{1}{\sqrt{detR}} \exp\left(-\frac{1}{2}(\alpha^{-1}A.(R^{-1} - I).A)\right) \quad \textbf{(10)}$$

where I is identity matrix, R^{-1} is the inverse correlation matrix and

$$A = \begin{pmatrix} \alpha^{-1}(V_1) \\ . \\ . \\ . \\ \alpha^{-1}(V_d) \end{pmatrix}$$

It is important to note that the correlation matrix R, in the Gaussian copula, is defined by the correlation matrix among the inverse univariate cumulative distribution functions. Hence the correlation matrix can be defined as (we have shown a two dimensional case that can be easily extended to multi dimensional)

$$R\left(\alpha^{-1}(V_1), \alpha^{-1}(V_2)\right) = \left(cov\left(\alpha^{-1}(V_1), \alpha^{-1}(V_2)\right) / \sigma\left(\alpha^{-1}(V_1)\right)\sigma\left(\alpha^{-1}(V_2)\right)\right) \quad \textbf{(11)}$$

Using the above equation we can calculate the correlation between two normally distributed variables. The value of correlation can be used to define the relationship among the dimensions.

A Gaussian copula has a simple procedure for its calculation and depends mainly on one parameter, the correlation matrix. Furthermore, Gaussian Copula depends on marginal analysis of the variables, the parameters defined as the inverse of marginal cumulative distribution functions, and the correlation between the inverse marginal cumulative distribution functions respectively. The complexity of the operations needed for marginal analysis of the variables and parameters are $O(k)$ for $\alpha^{-1}(.)$operation and $O(k^2)$ for R (parameter matrix). Thus, it is clear that for a low value of k (low dimensionality) and data for which marginal is easy to model, we can use Gaussian

copula to efficiently calculate the joint distribution of multidimensional vector. Our vectors are in a 31-dimensional space (which can be considered a low dimensional space) and the quantized values can easily be processed for marginal calculations. Hence, for our objective, the Gaussian copula seems to be a good fit. The quadratic computational time, with respect to k for calculation of correlation matrix, makes it easy to represent an image through its copula shape [27]. These points are the main motivation for choosing the Gaussian copula for our algorithm.

By using copulas, we have expressed the dependency structure among the multiple directions of our feature space. We calculated the correlations among different directions of the feature space. As mentioned earlier, the HOG features are 31 dimensional; hence, in total $31 \times 31 = 961$ correlation values will be calculated in order to define the dependency structure among all the directions of the feature space.

In the following discussion, we show the graphical representation of the data, from two randomly chosen directions of the HOG feature space, for a given partition of the image. Figure 7 shows the image for which we have calculated the HOG features and the sample distribution of the data for the two selected dimensions.

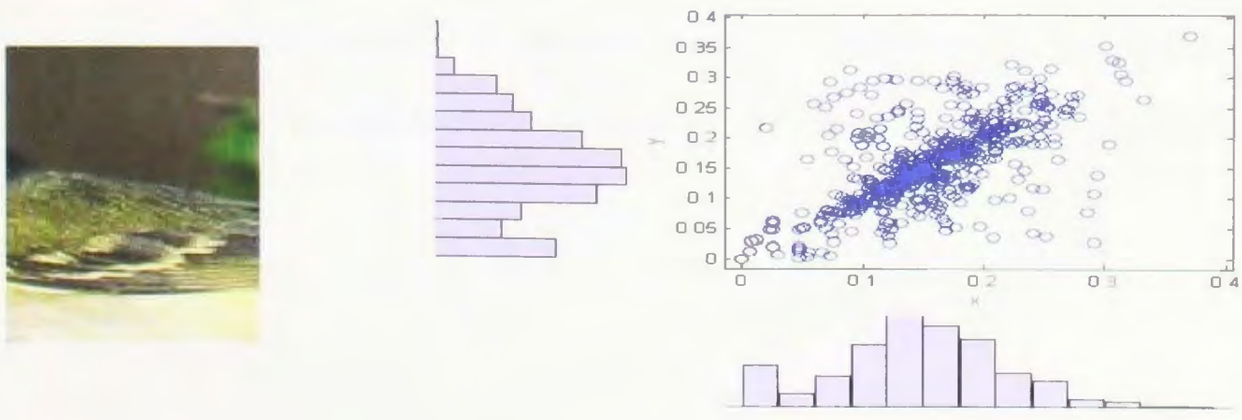


Figure 7: Back of a bird and the scatter diagram of its two dimensions of HOG features

By using the copula parameter, we calculated the dependency among these two dimensions. Using the calculated copula factor, we again generated random samples which are shown in the Figure below.

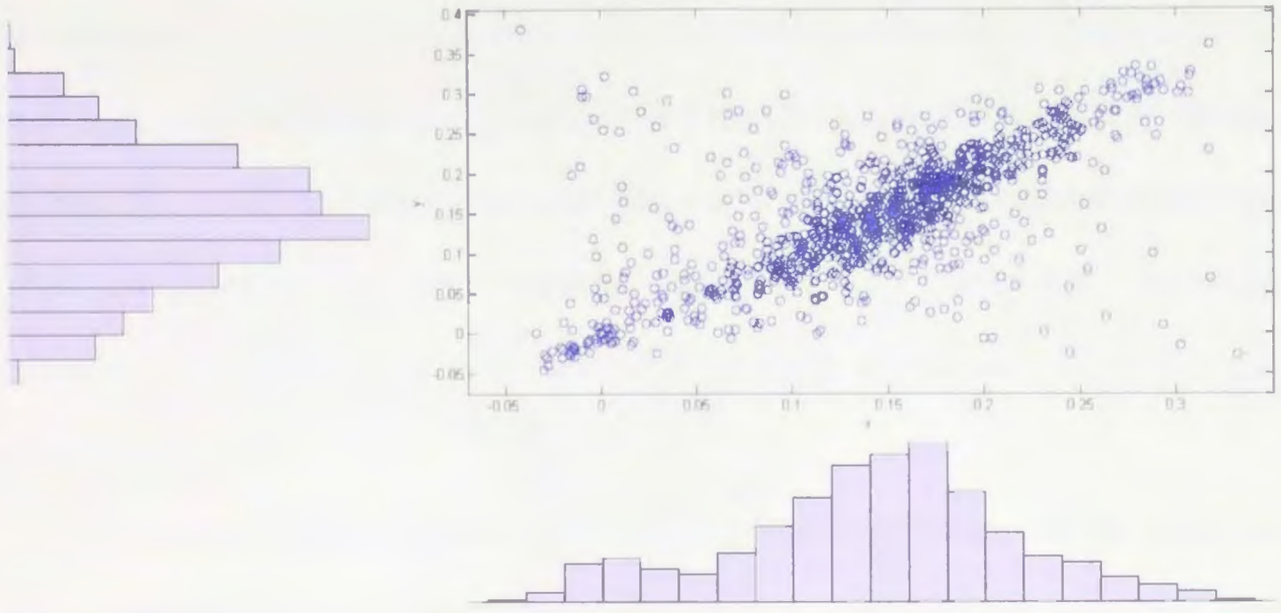


Figure 8: Random sample generated using Copula factor

From Figure 8 it is evident that copulas prove to be a good measure of dependency since the random computed points seems to follow the same path as the original graph. Copulas are used widely in the financial sector and risk analysis; however, most often in bivariate cases. Here we have tried to model the dependency structure for the multiple direction of the HOG features. A detailed description of the feature vector, and how we used it to encode information about the parts of our object, is given in next chapter.

3.3 Colour Histogram

Colour histograms are used to encode the colour information of the images. They represent the distribution of the colours from a defined colour space for each pixel. The most commonly used colour spaces are RGB (Red, Green and Blue) and HSV (Hue, Saturation and lightness). A histogram is a graphical representation of the number of pixels that fall in each particular type of colour within the image. In other words, we can say that the colour histogram represents the distribution of the colour composition. Colour histograms are used only to show the distribution of the colours within an image. It does not include any type of geometrical information, or information pertaining to spatial arrangements.

Usually the colour space in the images is quite large; hence, for the ease and effective representation, the colour space is divided into small intervals called bins. This process is called colour quantization. We then count the number of pixels in each bin to construct the colour histogram.

Colour histograms are a well-known concept in the area of image processing because they are simple and very effective. In our work we have included the colour information in our feature vector because colours are very important in the classification of the species. In a subordinate level classification, objects are of a similar shape with few, subtle differences; however, the colours may create a distinction amongst the species. We are not discussing the colour histogram in detail as it is well known concept for capturing the colours information in images.

CHAPTER 4 ALGORITHM

The solution proposed in this work has two stages. First is the training stage in which we train our system using given sample images for each species. Second is the classification stage in which each test image is classified into one of the training classes that are based on the extracted features and the model learnt in the training stage. Figure 9 shows the block diagram for the process of extracting the feature vectors from training images. The following subsections explain different steps of our algorithm.

4.1 Preprocess Stage

The very first requirement of this approach involves a pre-processing of the images. In this stage, we mark out the co-ordinate points for different visible body parts of the subject in the image. To do this, we have to identify the major body parts of our subject fish, so that we are able to point out those parts through coordinates on the image. We identified nine body parts namely Nose, Gill Cover, 'Eye', Pectoral Fin, Lateral Fin, Anal Fin, Caudal Fin, Back Body and Pelvic Fin for our test subject. The following Figure 10 shows a fish with these body parts annotated.

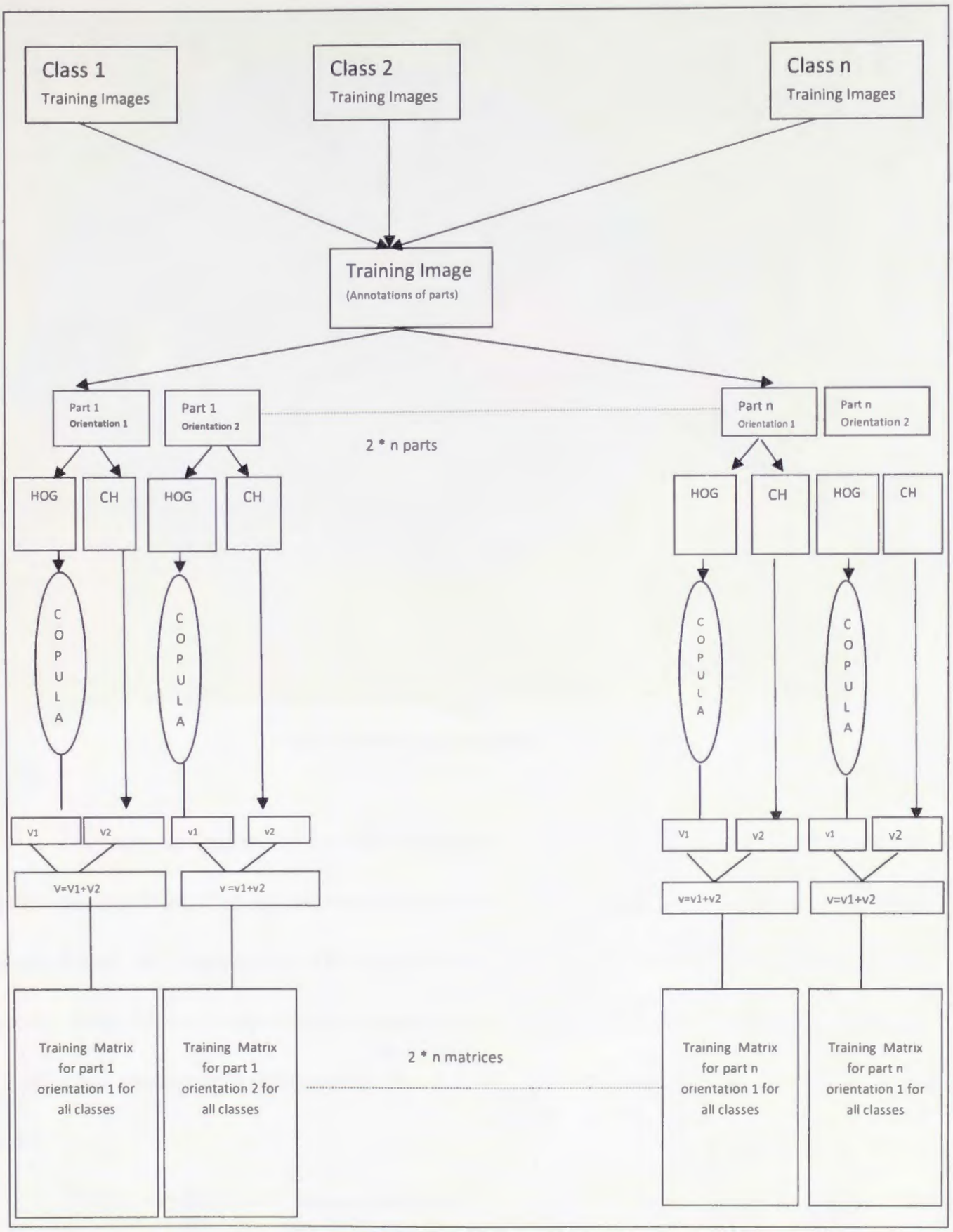


Figure 9: Diagram showing calculation steps for training matrices for cropped parts for both orientations

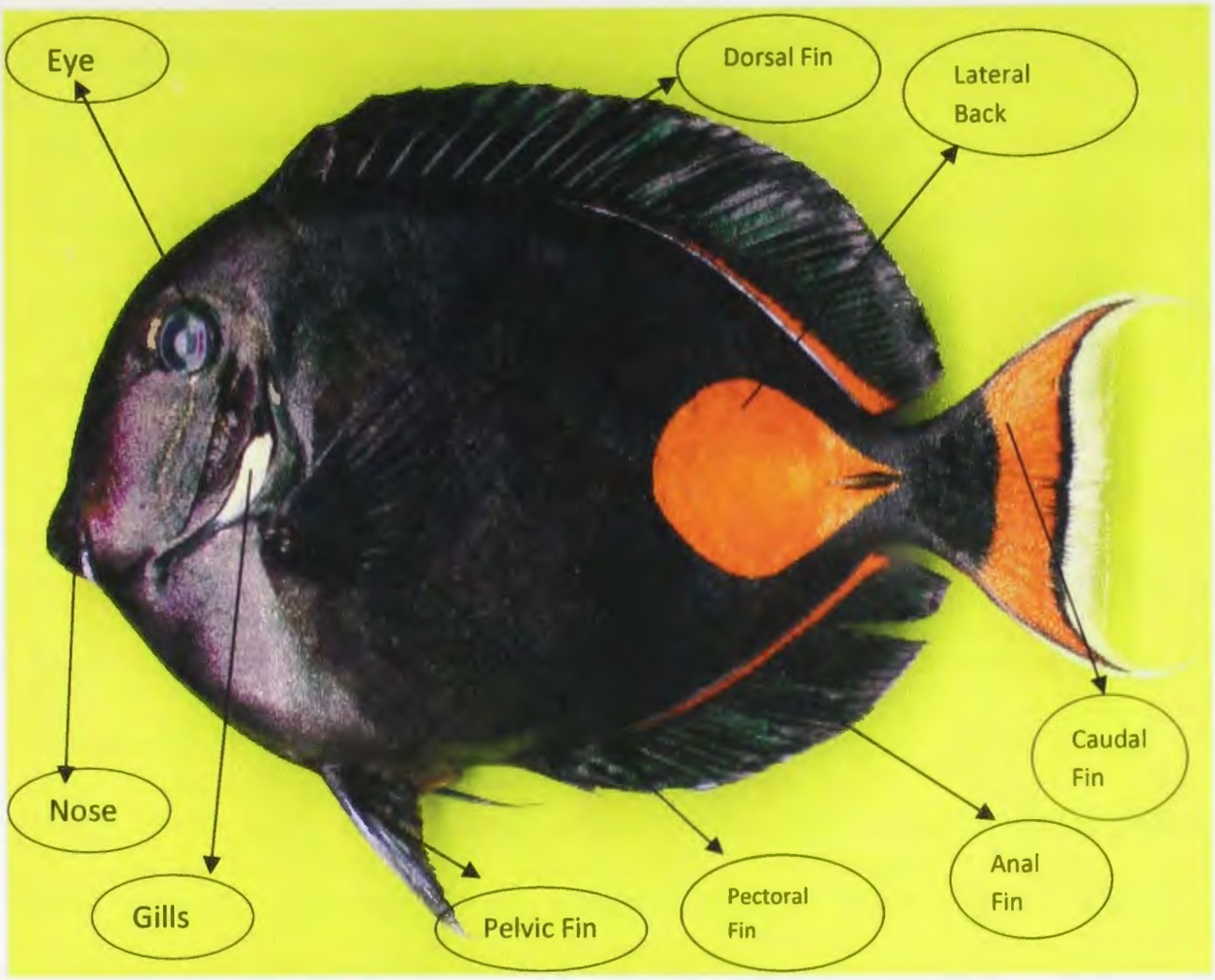


Figure 10: Body parts of fish

For each image, we record the co-ordinates of whatever body parts are visible in the image. By doing this for each and every image in our database, we record co-ordinate points for all the body parts. These co-ordinate points are used to crop a certain area around them. These cropped areas contain only one specific body part; thus, restricting the effect of background information. We follow the same procedure for all visible body parts.

These pre-processed images are used as inputs for our two main stages: The Training Stage and the Testing Stage. The following sub-sections cover these two stages in detail.

4.2 Training Stage

The training stage involves a series of steps that includes the cropping out of different parts of the subject, extracting the shape features, generating colour histograms, and then storing of feature vectors based on body parts *et cetera*. The following sub-section will examine each step.

4.2.1 Cropping of Body Parts

As discussed earlier, for subordinate level classification problem, focusing on local level features are expected to produce better results than comparing global postures. Considering this, we cropped out the pre-defined body parts of the subject from the image. In the pre-processing stage, we store the co-ordinates of the center points of the body parts (9 body parts for our subject). Using these co-ordinates, we crop out a rectangular section of the image while keeping the co-ordinate at the center of the rectangle, for each body part. We crop each body part into two orientations. In the first orientation, we keep the length at the X axis as 64 pixels and the Y axis at 32 pixels; for the second orientation we keep the length at the X axis as 32 pixels and the Y axis as 64 pixels. In both the orientations, the length and the width of the cropped rectangle are interchanged which keep the area of rectangle unchanged; however, the orientation gets differed. Choosing these two orientations helped us by covering the shape of each body part in a better way. This is easily visible in Figure 11 and Figure 12. Our research subject is fish; however, in this work we are also using figures for birds from the Caltech database in order to clarify the ideas. Figure 11 shows the cropped part of a sample fish image, from the fish database. Additionally, we have shown cropped part from a sample bird image, from CALTECH database, in Figure 12.

For each species, we have defined number of body parts as 9 and 15 respectively; however, in the Figures we have shown only a few body parts with views in both orientations. Please note that the cropped parts have been resized to fit in the figure and do not represent the actual cropped sizes. It is evident that choosing two orientations would help in getting more information around the selected body part. If we look at the anal fin and lateral fin part of the fish, the second orientation provides the more information than the first, because it shows a relative connection with the balloon structure (an essential structure on the body). In the case of the right wing of the bird, the first orientation provides better information than the second one, since it covers more structure of the bird's body. Our subject is broken down into different visible body parts (two orientations for each part) and then we separately extract the features for each cropped part. Feature matrices are stored separately, for the each orientation of every body parts. We look at our subjects in terms of their body parts rather than looking at their global posture.

Once we crop visible body parts from the images in both the orientations, the next step is to calculate the HOG features for each cropped part. The following discussion explains the parameters that we use to calculate the HOG features.

Orientation 1 and 2 respectively for Nose part

Orientation 1 and 2 respectively for Gill cover part

Orientation 1 and 2 respectively for Lateral Fin part

Orientation 1 and 2 respectively for Anal Fin part

Figure 11: Cropped body parts of fish

Orientation 1 and 2 respectively for Back part

Orientation 1 and 2 respectively for Beak part

Orientation 1 and 2 respectively for Belly part

Orientation 1 and 2 respectively for Right wing part

Figure 12: Cropped body parts of bird

4.2.2 Calculating HOG for each cropped part

For the calculation of the HOG features, we use the blocks of certain sizes to decompose the image (cropped part) into squared cells. Then the Histogram of Oriented Gradients is calculated for each cell. To calculate the features at different scales, we choose two sizes of blocks, 8x8 and 4x4, for the decomposition of the image. Figure 13 shows a block on the zoomed image. The image shown is of the anal fin of a fish.

Figure 13: A 8x8 block on the cropped image

Figure14:A 10 bin histogram of magnitude created for 64 gradients for a 8x8 block

We calculate the gradient vector for each pixel that falls within the cell. The gradient vector is one of the fundamental concepts in the area of image processing. It is a simple measure of change in the pixel values around a given pixel. It is measured in two dimensions: The X direction and the Y direction. For the change in the X direction, we calculate the difference in the values of the left and the right pixels, of the selected pixel. Similarly, for change in the Y direction, we calculate the difference in the values of the top and bottom pixels, of the selected pixel. We calculate the vector by putting these two values

together. The calculated vector will have its own direction and magnitude, using which; we can draw the gradient vector.

In the case of 8×8 block size, we have a total of 64 gradients. For each gradient vector we have magnitudes and orientations. We create a histogram of the gradient vectors by dividing them into histogram bins, based on orientations of the gradients. This process does the quantization of the gradient vectors. In the original version, the HOG features use a 9 bin histogram; however, Wang *et al.* [55] show that in the boosted HOG features, optimum performance is achieved by quantizing the vector into 10 bins. Since we are using the latter version, we use a 10 bin histogram for our calculation. We have used the signed and unsigned gradients; hence, the orientation varies from 0^0 *to* 360^0. Figure 14 shows a 10 bin histogram, created for an 8x8 block; the bins are of equal size and divide the 360^0 into 10 bins. The main reason for creating the histogram is to summarize the 64 vectors (2 components for each vector) into just 10 values (magnitude of bins). Additionally, this allows us to cope with slight variations within the orientations of the gradients, which may arise because of slight deformations. A slight deformation results in almost the same vectors; however, they may have slight difference on angles. The histogram does not store any geometrical information about the gradients, instead, it records the distribution information; hence it allows some ability to deal with any slight variations.

When creating the histogram, the gradients magnitudes are used. Using the magnitudes allows a better representation of the vector within the histogram; *i.e.*, stronger vectors will have a greater impact on the histogram. The magnitude of the gradients can be divided into two closest bins, when any of the gradients fall exactly on the boundary

between the two bins. For example, if the orientation of a vector is 72^0, then $\frac{1}{4}$ of the magnitude is added to the bin covering 36^0 *to* 72^0, and $\frac{3}{4}$ of the magnitude is added to the bin covering 72^0 *to* 108^0. The reason for splitting the magnitude into two bins is to minimize the problem of the gradients that sit right on the edge of the bins. The problem is that a slight change in the orientation of the gradient may cause the gradient to shift into the next bin that result in a huge difference in the resultant histogram. Distributing the magnitude between the two closest bins will reduce this sudden impact.

The next step in computing the features is the normalization of the histogram. The normalization of the gradients makes them invariant to any addition or multiplication (of any value) to the pixel value; *i.e.*, if one multiplies the pixel values by any given number, than it will have similar effect on all pixels, and the gradients calculated with new values will be the same. Similarly, an addition of any value to the pixel values will not affect the gradients after the normalization. In the same manner, the value of each bin increases by the same value, with which the pixel values are multiplied or added, if the histogram is not normalized. The normalization of histogram will make it invariant to any illumination change.

The histograms can be normalized by themselves; however, in this approach a block scheme has been used for normalization. In this approach, the histograms of the four cells are concatenated together with 40 bins (10 bins for each histogram). The resultant vector, from the concatenation, is divided by its magnitude to become normalized. While putting the blocks together, an overlap of 50% is allowed. It allows each single cell to appear different times in the final descriptor; however, on each appearance it has been normalized in respect to different cells. Figure 15 depicts this situation clearly. In Figure 16, the second block is created by concatenating the two

blocks from the previous block1 and than concatenate two new blocks. In block 2, we normalize the repeated blocks with respect to the data in newly added cells. Thus, even after repetition of the cells, a new view is added to the descriptor each time. In the original version, a descriptor is defined on a 36 dimensional space; however, in the latter faster version; the dimensions increase because of more bins to be quantized and also to accommodate the signed and unsigned vectors along with an energy feature. To keep the feature space in control, standard dimension reduction techniques are used to reduce the final reported dimensions to 31. The introduction of dimensions reduction techniques does not fall under the objective of this write up; further details can be obtained from the original paper.

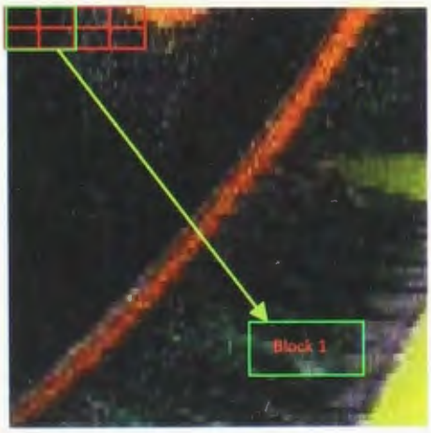


Figure 15: Creation of block from 4 cells

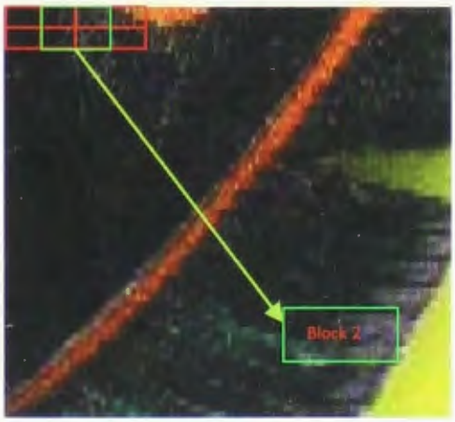


Figure 16: Creation of block from 4 cells with 2 cells overlapping

4.2.3 Copula- creating a dependency structure

In this step we find a dependency structure using the copulas among the different dimensions of the HOG features. As is already known, the version of the HOG features that we use has 31 dimensions. This 31 dimensional feature needs to be summarized in order to create a feature vector. The feature vector is created by using the correlation

values among the different dimensions of the HOG feature calculated using the copula theory.

By using copula theory, we aggregate the HOG features in order to develop a compact and effective image descriptor. For each dimension X^I, where I=1,....,31 represents the dimensions and, we define the marginal probability distribution $P_i(X^i)$ and cumulative distribution $CP_i(X^i)$, where i is either of the dimensions selected. If the joint probability is defined by $J(X)$, then we can approximate it by the set of marginal $P_i(X^i)$ and $CP_i(X^i)$ and a Gaussian function. Using Equation (11), we calculate the correlation matrix that gives us the correlation values among dimensions of the features. These correlation values are used to create a feature vector for every set of the HOG features. This process is used for every cropped body part.

For creating the image descriptor vector, we take the data of two dimensions at a time and calculate the correlation by fitting the Gaussian copula using the marginal. Thus, each dimension is processed with all other dimensions; hence, resulting in a total of 31x31 = 961 correlation values. These 961 values create the first part of our feature vector. Copulas have been extensively used in the area of finance and actuarial science for multivariate modelling because of their efficiency in estimating the joint probability in quadratic time [27, 29, 30]. Besides, HOG features have provided promising results in the human structure identification and in the object identification [23, 24, 55]. Inspired by these facts, we tried using both these algorithms to create an effective and efficient (small) image descriptor. To the extent of our knowledge, Copulas and HOG feature together have never been tried earlier when finding a solution for subordinate level problems.

4.2.4 Colour Histogram

The next step of our algorithm is to calculate the colour histogram. As discussed earlier, colours play an important role in the identification of species. In subordinate level classification problems, colours can become even more important because of same body structure of the subjects. One of the best and simplest ways to include the colour information is through a colour histogram. A brief description of the colour histograms has already been given in previous section.

The colour histogram purely focuses on the quantizing the colour information of the image; however, it does not give any geometrical information in its simplest form. We choose to calculate the colour histogram for our cropped body parts in order to retrieve the colour information pertaining to that respective part. We have chosen the RGB colour space for calculation. The main reason for selecting the RGB space is that it provides specific range of colours that are consistent across almost all platforms and it is widely used by the research community. The histogram is defined as a function over the colour space to approximate the counts of the pixels that have any of the possible colours.

For the calculation of the histogram, we decide to keep 6 bins per colour. The number of bins is an important parameter since the length of histogram is directly proportional to it. We tried to keep the number of bins low in order to have a small histogram. Later, we also try to experiment with increased number of bins per colour (7 and 8 bins per colour); however, no significant improvement is found. For the normalization of the pixel values, we use the $l1$ normalization scheme. Choosing the $l1$ normalization is just random. Since we have chosen 6 bins per colour for our histogram, the total length of our histogram is 216.

We calculate the histogram for both orientations since both the orientations are separately compared.

4.2.5 Feature Concatenation

We are trying to include the shape and the colour information of the body parts in our feature vector. We have encoded descriptors for both of them through the HOG features, encoded by the copulas and colour histograms respectively. Since we have both the encoded information, we concatenate them to form the final feature vector. The length of the final vector will be 961+216=1177, comprising 961 points of the copula structure of the HOG feature, and 216 points from the colour histogram. We have two feature vectors of the length 1177 for each orientation.

We store the feature vector separately for the two orientations of the each body part. For fish, we have 9 body parts and two feature vectors for the two orientations. We store part-wise features for both the orientations for all the training images. The resultant feature matrices are our training matrices.

Thus, following the above steps, we create the feature vectors for each visible body part in the training images and, store them in part-based feature matrices. We encode the shape and colour information for the body parts in the feature vectors. These vectors serve as the basis for the classification of the test images. The next step is the Test stage where we select a test image of a 'Fish' and classified it into one of the training classes using our classification algorithm. Figure 17 shows a diagram depicting the steps for calculation of our feature vector.

4.3 Test Stage

In the test stage, we classify the class of any test image, based on the comparison with the training matrices. The process follows the same steps as defined in the training stage. First we pre-process each test image to identify the part locations. Using the information of the part locations, we crop the image, using a fixed size predefined shape (fixed sized rectangle) centered at the given part location. We crop the parts into two orientations, similar to the way we did in the training stage, setting them as orientation 1 (rectangle of length 64 and width 32) and orientation 2 (rectangle of length 32 and width 64). In this way, we crop the visible parts of the subject in the test image. For each of the cropped parts, we calculate the HOG features on a block size of 8×8 and 4×4 in order to allow feature extraction on multiple scales. For each scale, we compute a Gaussian structure and calculate the correlation matrix among the different directions of the feature space. For each orientation, we calculate the colour histogram to include the colour information of the cropped part. We separately concatenate the two features for both orientations, for each cropped part. Each feature vector for the respective body part is compared with the training matrices of the same respective body part, for both the orientations.

For the comparison of the feature vector of the test image with the training matrices, we use the Euclidean distance method. This method shows the linear ordinary distance between the points in the Euclidian space, which is defined as the length of the line, connecting two points. The concept of the Euclidian distance method originates from the Pythagoras' theorem. This theorem is well known among researchers. Here, we

provide a simple formulation of Euclidian distance between the two vectors for the ease of understanding. It is easily extendable to multiple dimensions.

If there are two vectors $X = [x_1, x_2]$ and $Y = [y_1, y_2]$, for which we need to calculate the Euclidian distance, we need to have the squared differences within vector coordinates. Than we sum up the squared differences of the vector coordinates to get the squared distance of the vectors. The squared distance (SD_{xy}) can easily be shown from following equation :

$$SD_{xy} = (x_1 - y_1)^2 + (x_2 - y_2)^2 \quad \textbf{(12)}$$

The distance (D_{xy}) is defined by the square root of Equation (12) as shown below:

$$D_{xy} = \sqrt[2]{(x_1 - y_1)^2 + (x_2 - y_2)^2} \quad \textbf{(13)}$$

In the similar way we can extend the concept to multiple dimensions. In a Z dimensional space, if coordinates of two points X and Y are given by $X = (x_1, x_2, \ldots., x_z)$ and $Y = (y_1, y_2, \ldots., y_z)$ than the distance D between these two points is given by following equation:

$$D = \sqrt[2]{\sum_{i=1}^{Z} (X_i - Y_i)^2} \quad \textbf{(14)}$$

Equation (14) defines the Euclidian distance, which enables the notion of distance in a multi dimensional space.

We use the Euclidian distance measure to compare the test feature vector with the training feature matrices in order to classify the class. Here it has to be noted that we opt for a part based comparison of the feature vectors. This means we are comparing the feature vector of part 1 with the training matrices of part 1 and, the feature vector of part 2 with the training matrices of part 2 and so on.

Different parts of the same image may be classified into different classes. To decide on the final classification, we choose a simple voting scheme. We give one vote to each class that is ever classified for any part. In the last, the class with the highest vote is defined as the classified class for the chosen image. To understand let's say for part 1, class 2 is the classified class based on Euclidean distance; hence, we give 1 vote to class 2, similarly for part 2 and part 3 we find that the classified class is class1; hence, we give 2 votes to class 1. Similarly, for part 4, 5 and 6 the classified class is class 2; hence, class 2 now have 4 votes. Let's assume that the other 2 parts have been classified under class 5 and class 4 respectively, giving each of them 1 vote. After all parts have been classified under some category, we find that class 2 had the highest number of votes hence; it will be defined as the final classified class for the image. Figure 17 shows the test stage process.

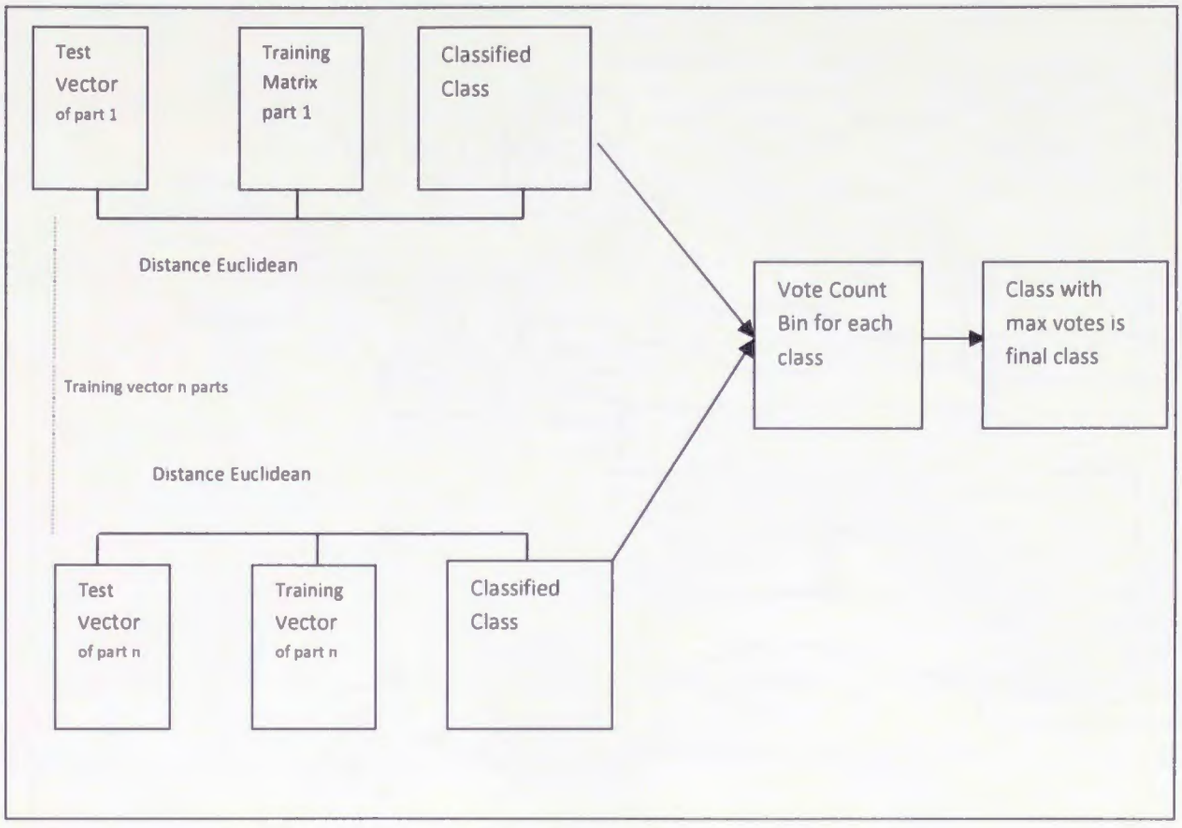


Figure 17: Diagram showing calculation steps for test stage

Figure 18 shows a block diagram that summarizes the whole steps of our algorithm.

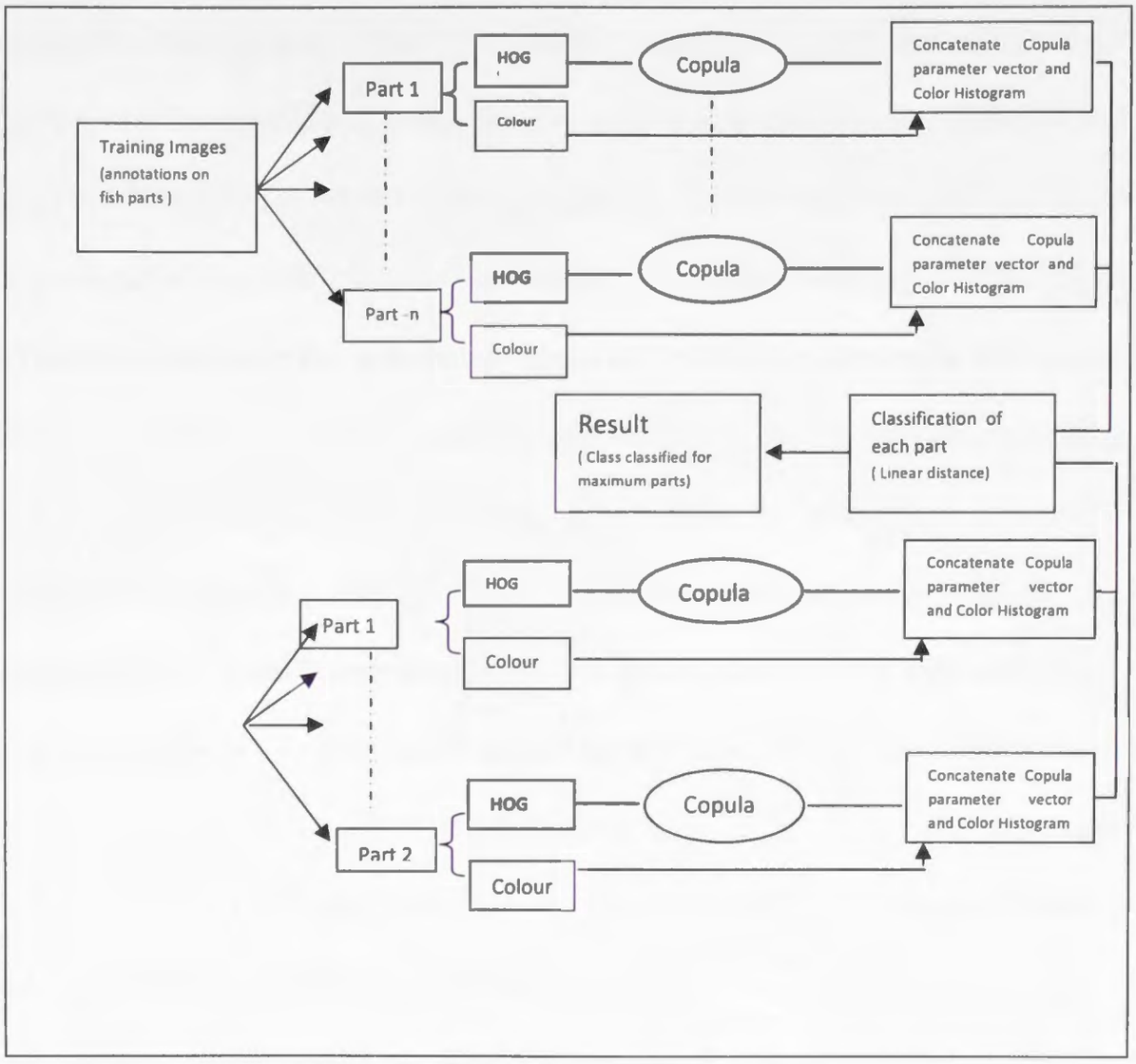


Figure 18: Shows the block diagram of the proposed approach

CHAPTER 5 TEST ENVIRONMENT

For evaluating the performance of our method, we need to find a proper database of fish images that have clear labelling of the species, represented in the image. Fish Base is a global information system about fish that is maintained by the World Fish and Food and Agriculture Organization of the United Nations [19]. This database has one of the largest collections of the fish pictures that include the proper labelling of the diffent species. They claim to have the information about all known fish species in this world. We select species that look alike in order to provide a challenge when discriminating amongst them. We do not preprocess the images to remove the background in order to keep things more challenging. The only problem with the existing database is that the sample pictures for any class is very low in number. For most species we have only 2 to 3 pictures. With this number of pictures, it is very hard to train the classification method. We try to find species, which are look alike, and that have acceptable number of sample pictures. We select the species that have 6 pictures for each category. We keep 3 pictures per class for training and use the other 3 pictures for testing.

To further test our method, we also experiment with The CALTECH USDS birds 2011 database. This database has a large number of the sample pictures for each species, and has predefined information about the different body parts of the subject in each image. This database has been prominently used for the experiments by the researchers in testing subordinate classification methods.

CHAPTER 6 RESULTS

As we have mentioned earlier, the sample 'Fish' pictures have been taken from the Fish Base [19], which has very few images for any species. The highest number of the images that we find for any category is 6. We find 10 different kinds of 'Fish' with 6 images for each category. Fortunately, the species that we select are very look alike, hence providing a good base to test the method. However, testing on such a small database does not give enough credentials to support the method. As a result, we also test this method against the standard sample [1, 2, 3], comprising images from 'Woodpecker' and 'Vireo' family, from the *'Caltech UCSD BIRD 200'* database.

Experiment Round No	Our Method (Accuracy %)	BOW Approach with Annotation (Accuracy %)
1	55.5	14.8
2	55.5	11.1
3	25.9	18.5
4	55.5	14.8
5	55.5	11.1
6	40.7	14.8
7	29.6	11.1
8	25.9	11.1
9	22.2	14.8
10	40.7	11.1
Average	40.7	13.32

Table 1: Classification Results for Fish database

Experiment Round No	Our Method (Accuracy %)
1	37.08
2	43.9
3	41.48
4	28.84
5	43.41
6	30.77
7	41.48
8	43.9
9	41.48
10	42.03
Average	39.44

Table 2: Classification Results for Vireo and Woodpecker family in Caltech Bird Database

To test the method against the diverse set of test and training images, we run our algorithm multiple times; each time a random selection for test and training image is done. Thus, in each round we have a fix number of images for test and training, however the images are shuffled in order to have different sets each time. The same process is followed for both the databases. In our method, we have simply defined the classification accuracy by following formula:

$$\text{Accuracy} = \frac{X}{Y} \times 100\,\%$$

where X = number of images whose true label has been correctly identified and, Y= total number of images.

The results of experiments, with our method, have been shown in Table 1, and Table 2, for the fish and bird databases respectively. The average classification accuracy for 10 rounds of experiments with the fish database is 40.7% and, the average classification accuracy for the sample bird database is 39.44%, for the same number of rounds.

The classification accuracy for the fish database is in the range of 22.2% (minimum) and 55.5% (maximum) whereas for the bird database it is in the range of 28.84% (minimum) and 43.9% (maximum). A possible reason for this variation lies in the fact that results are dependent on the number of parts that are visible in training images. It is possible that randomly chosen training images may have a certain part appear less often; hence, have relatively less training for proper classification of the respective part.

To the best of our knowledge and literature review, we do not find any work about fish species classification to which we can compare our results. Hence, we decide to do experiments, using the state of the art *'Bag of Words' approach* on our fish image

database, and compare the results. *'Bag of Words'* has performed very well with various complex databases such as Caltech 101 [56].

Our method has used annotation information for processing of the images. Hence, to do a fair comparison, we utilize the annotation information while experimenting with the Bag of Visual Words approach. Like our method, we crop different body parts of the subjects in images using the annotations; then we apply 'Bag of Words' approach on the cropped images. We create a separate library of visual words for each body part. Then, for the test images, we crop each body part using the annotations, and do a part-based comparison against the visual vocabulary of the respective body part. In this way, we have provided the same inputs to the 'Bag of Words' algorithm that we did to our algorithm. Finally, we check which class had been classified for the maximum body parts and, choose that as the final class for the test subject in the image.

The results for the 'Bag of Words' approach have been included in the Table 1, under the table column header 'BOW approach with Annotation (Accuracy%)'. Table1 displays a direct comparison of the results from our method and the 'Bag of Words' approach. It is clearly evident that our method is better than the latter. A possible reason for it might be the nature of classification, which is of subordinate level in our case, rather than being an object level classification. The BOW method uses a coarse coding which doesn't consider the finer details of the subjects and hence results in poorer performance in sub ordinate classification.

One of the famous databases in the area of subordinate level classification is the Caltech UCSD bird database. The database is very challenging because of the high degree of similarity amongst the subjects 'Birds'. They have much similarity with regards to

colour and shape. Most of the researchers [1, 3, 7] have used only a subset of 14 species (woodpecker and vireo family) out of the 200 species to produce their results.

We run our algorithm on the above subset, in order to check the performance of our method. We take the same standards for our experiment that the researchers [1, 3, 7] have taken for training and testing, and keep 15 images per category for the training and the rest for the testing.

We follow the same testing standards, in this experiment, that we follow with the fish database. We run our algorithm for 10 times, and randomly select the test and training images in every round. The results of these experiments are shown in Table 2. The best classification accuracy of 43.9% was achieved in round 2 and round 8, whereas the average accuracy is 39.44%. Our results are very much comparable with results of other researchers. Table 3 shows this comparison; it is clear that our method gives competitive results even on the Caltech UCSD bird database. Our average accuracy is just a little bit lower than that of Yao, Fei Fei [1] in which they have focused on the direct matching of templates, based on a very complex mechanism.

Respective Work	Results (%)
Lazebnik,Schmid and Ponce [10]	37.12
Steve, Catherine et-al [44]	37.02
Farrell, Oza et-al [3]	40.25
Yao, Bradski and Fei-Fei [1]	44.73
Ours	**39.44**

Table 1: Result Comparison with other reported work

From the results, we can say that our method for 'Fish Classification' is stable and holds firm to the classification task on subordinate level. We have kept the number of training images strictly to 15. If we increase the numbers by few, say 20 training images per category than results can further be improved.

We plot the ROC curves for the classification results of our method on the fish and bird database separately. They are shown in Figure 19 and Figure 20 below:

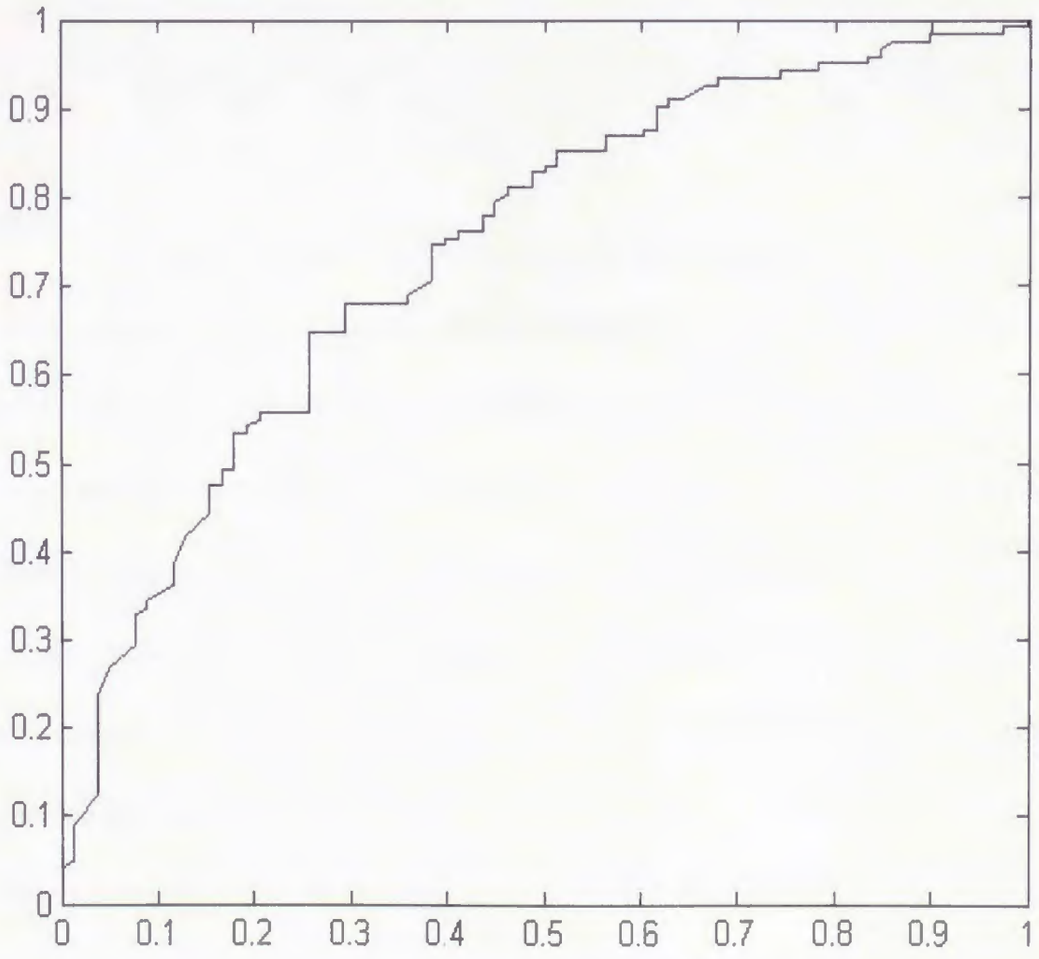


Figure 19: ROC curve for results on fish database

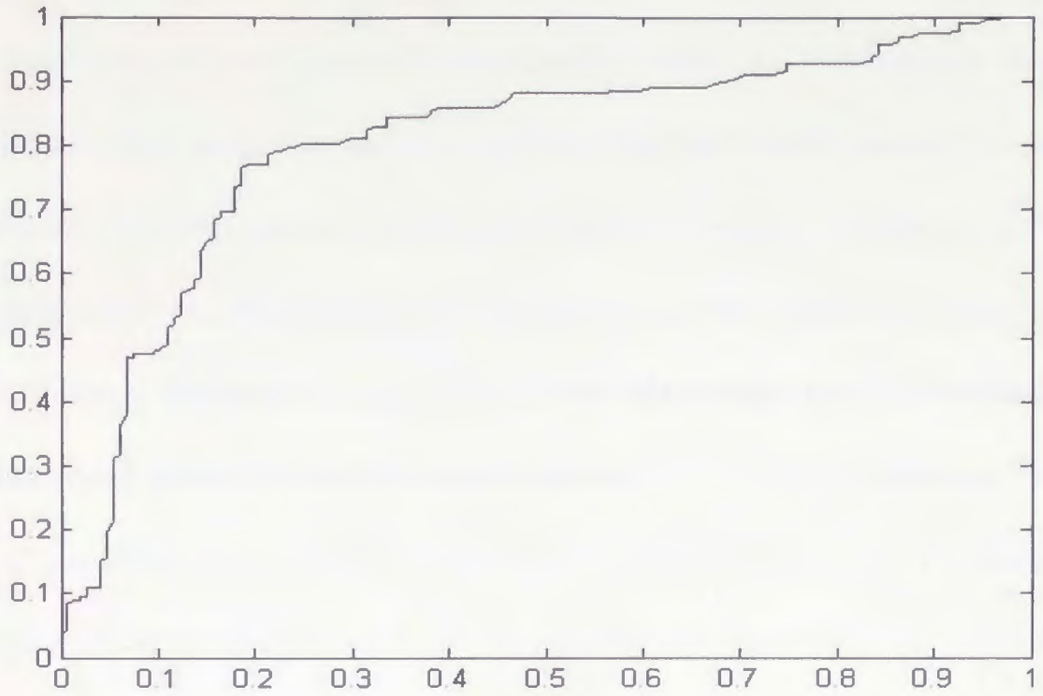


Figure 20: ROC Curve for results on Bird Database

ROC curve is the graph plot, which illustrates the performance of the classifiers. In the ROC curve; the sensitivity (true positive rate) is plotted against the false positive rate (1-specificity) for the different threshold settings. The accuracy of the classifiers is determined by the area under the ROC curve. A perfect curve always has the area 1; However, the curve that cover the area less than or equal to 0.5, shows a worthless test. For fish database, the area under the ROC curve is 0.7409 whereas for the bird database the area under the curve is 0.8039.

We would also like to mention that we tried to contact the other researchers in order to provide us support about their work; this was done so that we can check the performance of their method on our images, but no one replied. Hence we take the opposite approach, and run our method on their database in order to compare the performance of our method against their published results.

CHAPTER 7 DISCUSSION AND FUTURE WORK

We have proposed an annotation-based method for fish species classification using copulas. This work focuses on extracting the finer level statistic for subordinate classification. For this purpose, we extract features around the interested body parts of our subject 'Fish' by cropping the image sections of the respective body parts, in a predefined shape (rectangle in our case). For subordinate level classification, other researchers have primarily used the same approach, to extract the features from a sub-region of the image that includes the point of interest [1, 3, 6, 7]. We have used the colours and gradients to compare the image sections just like other researchers have used colours [1, 3, 7], gradients [1, 3, 7], texture [1, 3] and other basic features to compare their image sections.

Zhang *et al.* [3] have considered the object as a *'constellation of volumetric parts'*; they define their classification method, based on the shape and photometric appearance by creating a poselet comparison for the head and the body. In our method, we consider our subjects as a collection of parts and take 9 distinct body parts for a localized and finer level comparison. Yao, Khosla and Fei Fei [7] propose the use of rectangular patches to locate the region that comprises the body parts that are useful for fine-grained-Categorization. In a similar approach; Yao, Bradski and Fei Fei [1] have proposed the use of rectangular patches to generate random templates for comparing the different parts of the object. In essence, all of the major proposed works have focused the feature comparison on around the interested body parts for a fine-grained or subordinate level categorization. Our approach helps in identifying statistics at finer level, which keeps the

comparison only to the effective points. Furthermore, our approach helps in excluding the background information that is irrelevant in fine-grained-categorization.

The next important step, for a subordinate level classification, is the choice of features that should be used for making comparisons. It is obvious that objects have more or less the same body parts with subtle differences in their formation and colours. Yao, Bradski and Fei Fei [1] use a template matching approach in which they use the combination of colour and gradients features on a given location in the comparison of the templates. Farrell, Oza, Zhang *et al.* [3] use the properties of geometric shape for their classification work. Yao, Khosla and Fei Fei [7] use a combination of SIFT and colour information to compare the image patches. In this work, we have proposed the use of the Histogram of Oriented Gradients and the colour histogram, to catch the shape and colour information respectively for the target body parts.

In our feature vector we have encoded the information about the shape and the colour of the respective body part. We have focused on keeping the things simple by concatenating the vectors comprising the gradient information and the color information, and use linear distance method for comparison. This method is simple and does not involve the use of complex mechanisms and, achieves very good results. The method has been developed primarily for fish species classification; however, we also tested it against the Caltech bird database and found promising results. The major constraint with this method is its dependency on the annotations that it needs to crop the image section around the interested body part.

To the extent of our knowledge, this is the first attempt when copula theory has been tried to aggregate the multidimensional features (HOG) for the subordinate

classification. We achieve good results in defining a dependency structure among the dimensions of the feature space, using the copulas.

The major drawback of this method is its dependency on the annotations to locate the body parts. The future scope of this work might be the methods to avoid the annotation dependency. Besides, other gradient methods can be tried for feature extraction. Instead of colours other features such as textures and many more can be tried in combination with other features.

REFERENCES

[1] B. Yao, G. Bradski and L. Fei-Fei, "A codebook-free and annotation-free approach for fine-grained image categorization" in CVPR , Providence, 2012.

[2] I. Biederman, S. Subramaniam, M. Bar and J.Fiser, Subordinate level object classification reexamined, Psychol Res., 1999.

[3] R. Farrell, O. Oza, N. Zhang, V. I. Morariu, T. Darrel and L. S. Davis, "Birdlets: Subordinate categorization using volumetric primitives and pose-normalized appearance" in ICCV, Barcelona, 2011.

[4] A. Hillel and D. Weinshall, "Subordinate class recognition using relational object models" in NIPS, 2007.

[5] J. Sanchez and F. Z. Akata, "Fisher vector for *fine-grained* categorization" in CVPR workshop on FGVC, Colorado Springs , 2011.

[6] C.Wah, S. Branson, P. Perona and S. Belongie, "Multiclass recognition and part localization with humans in loop" in ICCV , Barcelona , 2011.

[7] B. Yao, A.Khosla and L. Fei- Fei, "Combining randomization and discrimination for *fine-grained* image categorization" in CVPR , Colorado Springs, 2011.

[8] E. Rosch, C. Mervis, W. Gray, D. Johnson and P. Boyes-Borem, "Basic objects in natural categories" Cognitive Science, vol. 8, no. 3, pp. 382-439.

[9] G. Csurka, C. Dance, L.Fan and J. Bray, J.Williamowiski, "Visual categorization with bag of keypoints" in Workshop on statistical learning in computer vision, ECCV, 2004.

[10] S. Lazebnik, C. Schmid and J. Ponce, "Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories" in CVPR, 2006.

[11] J. Wang, J. Yang, K. Yu and F. Lv, "Locality-constrained Linear Coding for image classification," in CVPR, 2010.

[12] M.-E. Nilsback and A. Zisserman, "A Visual Vocabulary for Flower Classification," in CVPR, 2006.

[13] G. Martinez-Munoz, N. Larios, E. Mortensen, W. Zhang, A. Yamamuro, R. Paasch, N. Payet, D. Lytle, L. Shapiro, S. Todorovic, A. Moldenke and T. Dietterich, "Dictionary-free categorization of very similar objects via stacked evidence trees," in CVPR, 2009.

[14] C. Wah, S. Branson, P. Welinder, P. Perona and S. Belongie, "The CALTECH USDS birds 2011," California Institute of Technology, Pasadena, 2011.

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

[16] Yu Kai, Z.Tong, Huang, S. Thomas, Xi Zhou, "Image Classification Using Super-Vector Coding of Local Image Descriptors," ECCV, 2010.

[17] L. Bourdev and J. Malik, "Poselets: Body part detectors trained using 3D human pose annotations," ICCV, 2009.

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

[19] "FishBase: A global information system on Fishes," World Fish and Food and Agriculture Organization of the United Nations, [Online]. Available: http://www.fishbase.org/home.htm. [Accessed 20 June 2015].

[20] D.G.Lowe, "Object recognition from local scale-invariant features," in ICCV , Kerkyra, 1999.

[21] X. Wang, T.X.Han and Y.Shuicheng, "An HOG-LBP human detector with partial occlusion handling," in ICCV, Kyoto, 2009.

[22] D. Lowe, "Object recognition from local scale-invariant features," in ICCV, Kerkyra, 1999.

[23] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in CVPR, San Diego, 2005.

[24] Q. Zhu, M.C. Yeh, K.T. Cheng and S. Avidan, "Fast Human Detection Using a Cascade of Histograms of Oriented Gradients," in CVPR, 2006.

[25] B. Wu and R. Nevatia, "Optimizing discrimination-efficiency tradeoff in integrating heterogeneous local features for object detection," in CVPR, 2008.

[26] P. Felzenszwalb, R. Girshick and D. McAllester, "Cascade object detection with deformable part models," in CVPR, 2010.

[27] M. Redi and B. Merialdo, "Direct modelling of image keypoints distribution through copula based image signatures," in International Conference on Multimedia Retrieval, Dallas, 2013.

[28] S. alZahir and F. Kashanchi, "A new image quality measure,," in 26th Annual IEEE Canadian Conference on Electrical and Computer Engineering, Regina, 2013.

[29] R. B. Nelson, An Introduction to Copulas, New York: Springer Series in Statistics, 2006.

[30] A. Sklar, "Fonctions de répartition à n dimensions et leurs marges," Publ. Inst. Statist. Univ., vol. 8, pp. 229-231, 1959.

[31] E. Rosch, C. Mervis, W. Gray and D. P. Braem, "Objects in natural categoris" Cognitive Psychology, vol. 8, no. 3, pp. 382-439, 1976.

[32] K. Mikolajczyk, C. Schmid, "An affine invariant interest point detector," in ECCV, 2002.

[33] T. Zang, Y. Gong, K. Yu, "Nonlinear learning using local using local coordinate coding," in Advances in Neural Information Processing Systems, 2009.

[34] A. Coates and H.Lee, Y. Andrew, "An Analysis of Single-Layer Networks in Unsupervised Feature Learning," in International conference on artificial intelligence and statistics , 2011.

[35] L. Jia. Li, S. Hao P. Eric. Xing and Li. Fei-Fei, "Object bank: A high level image representation for scene classification and semantic feature sparisification," in NIPS, 2010.

[36] S. Maji, L. Bourdev and J. Malik, "Action recognition from a distributed representation of pose and appearance," in CVPR, 2011.

[37] L. Torresani, M. Szummer and A. Fitzgibbon, "Efficient object category recognition using classemes," in ECCV, 2010.

[38] D. Hoiem, A. Efros and M. Hebert, "Automatic photo pop-up," in SIGGRAPH, 2005.

[39] P. Felzenszwalb, R. Girshick, D. McAllester and a. D. Ramanan, "Object Detection with Discriminatively Trained Part Based Models," in JAIR, vol. 32, no. 9, pp. 1627-1645, 2007.

[40] L. Bourdev, S. Maji, T. Brox and J. Malik, "Detecting people using mutually consistent poselet activations," in ECCV, 2010.

[41] Amazon, "Amazon mechanical turk," Amazon, [Online]. Available: http://www.mturk.com. [Accessed 21 June 2015].

[42] S. Hinterstoisser, S. Holzer, C. Cagniart, S. Ilic, K. Konolige, N. Navab and V. Lepetit, "Multimodal Templates for Real-Time Detection of Texture-less Objects in," in ICCV, 2011.

[43] C. Wah, S. Branson, P. Perona and S. Belongie, " Multiclass recognition and part localization with humans in the loop," in ICCV, 2011.

[44] B. Steve, W. Catherine, S. Florian, B. Boris, P. Welinder, P. Pietro and B. Serge, "Visual recognition with humans in the loop," in ECCV, Heraklion, Crete, Greece, 2010.

[45] A.Vedaldi, V.Gulshan, M.Varma and A. Zisserman, "Multiple kernels for object detection," in ICCV, 2009.

[46] P. Felzenszwalb, "A discriminatively trained, multiscale, deformable part model," in CVPR, 2008.

[47] L. Bourdev, S. Maji, T. Brox and J. Malik, " Detecting People Using Mutually Consistent Poselet Activations," in ECCV, 2010.

[48] G. Mart´ınez-Munoz, N. Larios, E. Mortensen, W. Zhang and A. Yamamuro, "Dictionary-Free Categorization of Very Similar Objects via Stacked Evidence Trees," in CVPR, 2009.

[49] E. Gal, Copulas in Machine Learning, Springer, 2013.

[50] X. Guo, L. Wang, J. Zeng and X. Zhang, "Codebook design algorithm based on copula estimation of distribution algorithm," in Robot, Vision and Signal Processing , 2011.

[51] V. Krylov, G. Moser, S. Serpico and J. Zerubia, "Supervised high-resolution dual-polarization sar image classification by finite mixtures and copulas," Selected Topics in Signal Processing, IEEE Journal of, vol. 5, no. 3, pp. 554-566, 2011.

[52] M. Redi and B. Merialdo, "Marginal-based visual alphabets for local image descriptors aggregation," in ACM International Conference on Multimedia, 2011.

[53] T. S. Durrani and X. Zeng, "Copula based Divergence Measures and their use in Image Registration," in 17th European Signal Processing Conference, Glasgow, 2009.

[54] G. Mercier, S. Derrode, W. Pieczynski and M. Nicolas, "Copula-based Stochastic Kernels for Abrupt Change Detection," in Geoscience and Remote Sensing Symposium, Denver, 2006.

[55] Z. Wang, Y. Jia, H. Huang, and S.Tang, "Pedestrian Detection Using Boosted HOG Features," in Intelligent Transportation Systems, Beijing, 2008.

[56] L. Fei-Fei, R. Fergus and P. Perona., "Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories," in CVPR, 2004 Workshop on Generative-Model Based Vision. 2004.