IMAGE CLASSIFICATION USING INVARIANT LOCAL FEATURES AND CONTEXTUAL INFORMATION

RAMESH BHARATH

(*M. Sc., National University of Singapore, Singapore*)

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

NATIONAL UNIVERSITY OF SINGAPORE

2015

DECLARATION

I hereby declare that this thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

Ramesh Bharath

13 August 2015

Acknowledgments

First and foremost, I would like to express my deepest gratitude and appreciation to my main supervisor, Prof. Xiang Cheng, for his immense dedication, encouragement to challenge the status quo, and emphasis on critical thinking throughout my graduate studies. It goes without saying that this thesis cannot be finished without his careful guidance, constant support and encouragement.

I would also like to express my great appreciation to my co-supervisor, Prof. Lee Tong Heng, for his guidance and encouragement throughout the past four years.

I would like to thank Prof. Cheong Loong-Fah and Prof. Yan Shuicheng for their kind encouragement and constructive suggestions, which have improved the quality of my work. I shall extend my thanks to all my colleagues at the Control & Simulation Lab and to the larger NUS community, for their kind assistance and friendship during my stay at National University of Singapore. Special mention has to be given to Arrchana Muruganathan, Arun Shankar Narayanan, Dr. Huang Deqing, Kyaw Ko Ko Thet, Manukumara Manjappa, Muthu Karuppan, Omid Geramifard, Prashant Shuvan, Rui Huang, Sudheer Vanga Kumar, Tan Yan Zhi, Yang Shiping, Yang Yue, and Xuilei Niu, for constant support and encouragement throughout our interactions.

On a more personal note, I would like to express gratitude to my school teachers, Ms. Padmini Iyer (#padminiiyerfanboy) and Mrs. Sundaravalli, whose selfless dedication inspires me to this day to be in academia. Speaking of teachers, I

would like to pay homage to the man who unbarred the gates to fearless inquiry into the rigid authority of the Vedas, and eventually left an indelible influence on Indian psyche ever since. Drawing inspiration from the skeptical philosophical movements of ancient India, the historical Buddha kept clear of all notions of selfhood and discovered extremely important truths about the nature of consciousness (Harris, S. (2014). Waking up), which eventually became mired in religious dogmatism. Besides reduced stress & anxiety levels, most of what constitutes as innovation in this thesis, is a direct result of having an open attitude towards the present moment, in line with the practice of mindfulness.

Finally, very special thanks go to my parents, Padmavathi Ramesh and Ramesh Ramanujam, for their everlasting love, support, and understanding over the years. I would also like to thank my younger brother, Raghav Ramesh, for taking care of our parents while I have been away in Singapore. Last but not least, special thanks to my best buddy and wife, S. V. Synthia, for her critical eye, support and encouragement.

Contents

Summary

Object recognition has been a central task to the computer vision community since the early days of using computers to identify hand-written characters. Through these fruitful decades of increasing machine intelligence, we have taken huge strides in solving specific tasks, such as classification systems for automated assembly line inspection, hand-written character recognition in mail sorting machines, bill counting and inspection in automated teller machines, to name a few. Despite these successful applications, computers have made little progress in generalizing object appearance, even under moderately controlled sensing environments. On the other hand, humans can effortlessly categorize hundreds of objects present in highly complex scenarios. We believe this success in pattern recognition is due to the variety of cues utilized by the human vision system. Therefore, the central topic is this thesis is a cue-based approach to object categorization.

There are several cues that assist, both humans and computers alike, in identifying objects from two-dimensional images. Primary among these cues is the shape of the object. The first contribution of this thesis is to propose a novel local shape descriptor using log-polar transform, which is robust to arbitrary scale, rotation and view-point changes. The proposed local feature based shape classification framework was tested on a widely used and challenging shape dataset with excellent improvement compared to existing works.

Secondly, we extend our binary shape classification framework to the more general case of classifying grayscale images. The second contribution of this the-

sis is then to develop a novel log-polar encoding of grayscale appearance cues, such as texture and structure, and binary shape information for classification of grayscale object images. The proposed image classification system was tested on the popular ETH-80 dataset with significant improvement in classification performance compared to state-of-the-art methods. Thirdly, we also demonstrate high classification performance on popular benchmark datasets, such as the Caltech-101 and Flickr-101 object dataset, using a novel multi-cue object representation of color images.

Finally, besides the above research works, we develop a real world application based on log-polar transform for monitoring vehicles on expressways. The novelty of this design is the usage of multiple depth-of-field information for tracking expressway vehicles over a longer range, and thus provide accurate speed information for overspeed vehicle detection. A novel speed calculation algorithm was designed for the composite vision information acquired by the system. The calculated speed of the vehicles was verified using RADAR speed detection systems and smartphone applications.

List of Tables

List of Figures

Chapter 1

Introduction

Object recognition has been a central task to the computer vision community since the early days of using computers to identify hand-written characters [1]. Through these fruitful years of increasing machine intelligence, we have taken huge strides in solving specific tasks, such as classification systems for automated assembly line inspection [2], hand-written character recognition in mail sorting machines [3], bill counting and inspection in automated teller machines [4], to name a few. Despite these successful applications, computers have made little progress in generalizing object appearance, even under moderately controlled sensing environments.

Many mammals, especially humans, perceive the world using visual cues as their dominant source of information [5]. Consequently, humans can effortlessly categorize hundreds of objects present in highly complex scenarios, which is made possible by the highly evolved visual cortex that accounts for a variety of visual cues. Therefore, we believe a cue-based approach to object categorization is key to achieving real progress toward intelligent systems, and this thesis aims to take

a step in this direction.

Several visual cues assist, both humans and computers alike, in identifying objects from two-dimensional images. Some examples are shape, depth, motion, texture, color, and 3D pose. Among these cues, shape is an elementary aspect of visual processing as it provides important clues about the identity and functional properties of the object. Hence, object recognition research in its budding years was primarily concerned with 3D shape representation [6, 7]. In the late 80's, the theory of recognition-by-components [8] proposed a powerful set of regularizing constraints using shape primitives for object recognition. It proposed that humans made use of easily detectable perceptual properties (curvature, collinearity, symmetry, parallelism and cotermination) that are invariant to orientation changes, distortion, and occlusion. Nevertheless, this theory has not been used successfully in natural images due to the representational gap between low-level features and abstract nature of model components. Subsequent two decades of research in object recognition moved away from 3D geometry to appearance-based recognition systems, which opened up new horizons in recognizing natural images [9].

Appearance-based recognition methods can be divided into global feature methods and local feature methods. The latter gained momentum in the first half of the 2000s mainly due to its superior performance in scenarios like clutter and partial occlusion [10]. The principal idea behind these methods is to extract several local features from an image and then identify the likely object from which those features were extracted. One of the pioneering local descriptors, scale invariant feature transform (SIFT), was developed by Lowe [11]. He de-

scribes an object recognition system that uses heuristically derived local features that are proposed to be invariant to image scaling, translation, rotation, and partially invariant to illumination changes [11]. Similarly, simple rigid template approaches with clever crafting of local features have also shown excellent performance [12]. However, heuristically designed feature descriptors fail to achieve invariant properties theoretically. Therefore, one of the focuses of this thesis is to develop a local descriptor with a sound mathematical basis for scale and rotation invariance.

To summarize this thesis, we begin with the investigation of binary shape image classification using local features invariant to scaling and rotation, and then investigate the classification of grayscale and color images by incorporating more local object cues. Finally, we present a video processing application based on the image sampling technique extensively used in this thesis.

A brief review of the recent results and related works are presented in this chapter.

1.1 Binary Shape Classification

Classification of shapes irrespective of scale, rotation, position and other appearance variations is a challenging and important problem in pattern recognition. While some progress has been made in resolving these challenges, shape classification has already found its application in numerous ad hoc machine vision settings [2] like assembly line inspection, surface corrosion detection, railroads parts inspection, laser butt-joint welding, wrist watch quality detection, to name a few. Over the past few decades, dozens of feature descriptors have

been engineered for shape analysis and classification [13–25].

One of the classical approaches for shape representation is to obtain 'shape invariants'. The idea is that one could compute functions of geometric primitives of the image that do not change under different image formation conditions and viewing geometry. Common shape invariants include (a) simple geometric invariants such as the cross-ratio, distance ratio, angle, etc.; (b) algebraic invariants such as determinant, eigenvalues [26]; (c) differential invariants such as curvature, torsion and Gaussian curvature [27]. However, shape representation using invariants has some major problems. First, shape invariants are usually derived from the pure geometric transformation of shapes, which are less applicable to non-rigid objects considered in this work. Moreover, invariants are very sensitive to boundary noise and occlusions [27]. Finally, the most challenging aspect of invariant methods is the matching using some form of subgraph method, which is known to be an NP-complete problem [26].

In order to overcome the above limitations of differential invariants, invariants based on integral computations have been proposed. One major drawback of integral invariants is that they are mostly global descriptors, and are thus sensitive to occlusion. On the other hand, recent works like the multiscale integral invariants [28] have developed local descriptors, which have been shown to have competitive performance for shape matching. However, multiscale integral invariants are invariant only to translation, rotation, and uniform scaling. Our work aims to achieve invariance even under non-uniform scaling by representing shapes structurally without assuming any geometric information.

In general, there are two ways of representing shapes to obtain a global or

structural descriptor: contour-based and region-based representation. Contour-based representations [19,20] extract information only from the shape boundary, whereas region-based descriptors such as image moment invariants [24], Zernike moments [25], shape matrix [23] analyze the shape as a whole. These two approaches can also be classified as *spatial domain* and *transform domain*, depending on whether the features are obtained in the spatial domain or in the transformed space. In practice, the spatial domain approach has been found to be very sensitive to noise, distortions and occlusions [27]. Out of the transform domain methods, Fourier transform based spectral analysis has been identified as a superior tool to represent shape for both contour-based and region-based descriptors [27]. Based on this line of reasoning, we choose to represent the binary shapes in the spectral domain using Fourier transform.

Global approaches create a holistic representation of the shape, and therefore, they are susceptible to corruption when there is a considerable viewpoint change or occlusion. On the other hand, local shape descriptors employed structurally, such as shape context [29], have been shown to be robust to deformations. It is to be noted that shape context creates log-polar histograms[1] instead of using the classic LPT, which is sampling the image at the intersection of rings and wedges of the transform. Moreover, the shape context requires a point-by-point matching scheme for two shapes, which makes it unsuitable for fast online shape matching [27]. This motivates us to employ the classic log-polar transform (LPT) [32] as a local descriptor, which converts scale and rotation changes in the image domain to horizontal and vertical translations in the log-polar domain, respectively. Therefore, by obtaining the Fourier transform modulus (the magni-

[1]Similar trend for grayscale images; popular examples of log-polar histograms are [12,30,31]

tude of the 2-D Fourier transform) of the log-polar sampling, scale and rotation invariance can be enforced. Although LPT has been used to obtain scale and rotation invariance in many computer vision applications (image registration [33], shape classification [32], grayscale object recognition/detection [31,34–36], image tracking [37], pose estimation [38]), it has been rarely applied to shape classification since the advent of powerful classification schemes such as the bag-of-words model.

The bag-of-words model has recently emerged as the dominant framework in image classification tasks, such as object and scene classification [39–42]. First, keypoint detection [39,43] or dense sampling [44,45] is done on the image to select patches of interest, followed by a description of each patch using SIFT [39, 46], raw patch [43, 47] or filter-based representations [44, 48]. Subsequently, the descriptors are quantized using a visual vocabulary or codebook that is commonly built using K-means [39, 44]. Finally, the histograms of the training images are used to train a linear/non-linear classifier. The bag-of-words framework was applied to shape classification with some success [49], which motivates us to employ it in this work.

The major disadvantage of the bag-of-words framework is the lack of spatial information in the histogram representation. This problem was alleviated by the introduction of spatial pyramid matching (SPM), which divides the image into increasingly finer regions and constructs a histogram for each region [50]. This results in a histogram representation with a dimension equal to the number of regions times the codebook size. Spatial pyramid matching has been widely applied to scene classification tasks and it is also responsible for inspiring an array

of works for the feature pooling step [51–54]. In general, higher classification accuracy has been linked to a larger vocabulary [50, 55], but saturation can be expected at some point [55]. In light of this fact, the histogram obtained from the SPM approach using a large codebook is very high-dimensional (21 times the codebook size for the standard 1×1, 2×2 and 4×4 representation), which compromises on training time and classification accuracy due to the 'curse of dimensionality' problem [51]. Therefore, the importance of contextual information, i.e., the spatial relationship between the local features has been explored by many researchers.

In the face of occlusion, noise, and variations in pose, several object categorization models use appearance and contextual information to improve classification accuracy [56]. The Markov stationary features (MSF), first proposed in [57], provides an interesting alternative for encoding spatial information by using the spatial co-occurrence matrix [58]. Although the stationary distribution is a unique method to extract features, it requires calculation of higher powers of matrices (typically 50) which can be extremely prohibitive for large codebooks. Moreover, the stationary distribution is an indirect method to capture information from the spatial co-occurrence matrix. In order to find an intuitive, yet a computationally less intensive way to encode contextual information, we consider the image as an article written using many "visual" words in the bag-of-words framework. Therefore, the problem of image processing is similar to language processing. In the domain of natural language processing (NLP) [59], which gave birth to the bag-of-words representation, contextual information is commonly incorporated using the N-gram model for text classification. Inspired

by this idea, we interpret each entry in the spatial co-occurrence matrix as a bi-gram count. Although interpreting the spatial co-occurrence matrix as a bi-gram count is not a new idea [60], we propose a novel method to extract bi-grams using the corresponding transition matrix. Besides improving the histogram representation in the bag-of-words model, choosing the codebook size and selection of local feature parameters also play a vital role in obtaining high classification rates. The following paragraph discusses these issues.

There are two very important considerations while using the bag-of-words model: the extracted local features and the codebook size. Most methods in the literature use a codebook size deemed to be large enough, simply chosen by trial-and-error, without using a solid criteria. However, there are a handful of recent works in the literature [61–64] addressing the problem of codebook size selection. In [63], an iterative method was designed for obtaining a codebook by merging two clusters that have minimum loss of mutual information. The input to the iterative method is a codebook generated by K-means, and thus inconveniently requires selecting a 'good' size in the first place. Recently, [62] reformulated codebook generation in a supervised setting as a neural network model. Note that the focus of this thesis is limited to unsupervised codebook generation in the traditional bag-of-words framework. In reference [61], conditional entropy and purity were proposed to evaluate the quality of the generated codebook. However, both these measures suffer from over-fitting, and therefore prefer arbitrarily large codebook sizes. As the number of clusters increases, purity and entropy reach their ideal values at the cost of having each sample as a cluster. A similar problem was encountered in the training of decision trees and gain

ratio [65] was subsequently introduced for selecting an optimal attribute. We take inspiration from gain ratio and propose a metric for choosing an appropriate codebook size in the bag-of-words model. Additionally, we propose an iterative method to jointly tune the codebook size and the local feature parameters using the training data.

To summarize, our focus in this thesis is to investigate and develop a robust classification framework for binary shapes that have scale, rotation and strong viewpoint variations.

1.2 Grayscale Image Classification

A cue-based approach to object classification is important for generalization to unseen objects. However, this aspect has been rarely studied due to the nature of training and testing protocol used for several grayscale image datasets. While the practice of using a random training and testing split avoids the bias of having a fixed training set, it leads to difficulties in objectively evaluating whether the training images yield a visual world model that can generalize to unseen objects of a known object category. Moreover, a significant obstacle for rigorously evaluating both appearance and shape based methods is the widespread use of databases without segmentation ground truth for the object categorization task. We address both these problems by adopting the rigorous leave-one-object-out cross validation protocol on the ETH-80 dataset, which provides segmentation ground truth for each object.

Several works extract different local descriptors, and treat them as different cues in their object recognition framework. For instance, [66] combined shape

cues obtained from SIFT descriptors and color cues obtained from the histogram of RGB values for object classification. Similarly, [67] ambitiously combined multiple interest point detectors and multiple descriptors for detecting objects in an image. Likewise, [68] combined dense SIFT, self-similarity descriptors, and geometric blur features with multiple kernel learning to obtain the final image representation. A similar attempt was made in [69] and [70] to combine multiple feature channels for image classification.

Differing from the above works, a handful of attempts have been made in the past with the aim of encoding multiple cues by designing a novel image processing method for object recognition. Reference [71] combined texture cues obtained from texture-layout filters [72] and contour fragments [73] obtained using sets of edges matched to the image using the oriented chamfer distance. In the same vein, [74] combined outline contour and the enclosed texture in pictorial structures for object detection. While popular descriptors like SIFT capture texture and gradient information, they do not explicitly encode shape information. However, there are a few handcrafted local shape descriptors, such as the pyramidal gradient descriptor [75], which is a histogram of oriented gradients computed on the output of a Canny edge detector. Similarly, some works [73, 75] do obtain local contour fragments to encode shape information from grayscale images. Although appearance based approaches have taken the forefront of object categorization research [10], shape based object categorization in natural images has been of increasing interest lately [76], with the help of advances in contour detection [77]. This thesis aims to take a further step by encoding grayscale texture, structure, and object shape extracted using saliency detection [78] in a

unified bag-of-features framework using log-polar transform.

Our work aims to achieve scale and rotation invariance, and in this regard, is most similar to [36], which has used the classic log-polar transform to achieve scale invariance without scale selection for grayscale images. Reference [36] presents scale invariant descriptors (SIDs) that use a logarithmic sampling on band-pass filtered images. As a result of the non-uniform scale of spatial sampling, centered at each pixel of the image, the authors showed that it is possible to obtain feature vectors that are scale and rotation invariant, by transforming the corresponding log-polar sampled amplitude, orientation and phase maps into the Fourier domain. In comparison to [36], we sample the shape boundaries of the extracted binary shape image by using the log-polar transform followed by obtaining its Fourier transform modulus. In addition, we also sample the structure and texture images on keypoints selected using an image denoising method, as discussed below.

Existing works have adopted two main strategies for selecting keypoints: (1) the simple but counter-intuitive strategy of densely sampling the entire image regardless of object boundaries, and (2) the more principled approach of designing sophisticated scale-and-affine invariant keypoint detectors. Our work takes a different approach for selecting keypoints, based on the assumption that a keypoint only needs to be visually salient with respect to its neighbors, and it need not possess invariant properties. Therefore, dealing with noise is a crucial aspect of such a strategy. In this regard, the most related work is in the image denoising literature, which has a multitude of algorithms reviewed extensively in [79]. Gaussian smoothing, anisotropic smoothing (mean curvature

motion), total variation minimization, and the neighborhood filters are examples of image denoising methods. Inspired by the success of variational methods on state-of-the-art optical flow benchmark datasets [80, 81], we choose the Rudin-Osher-Fatemi (ROF) model [82] to perform image denoising. In fact, the optical flow literature has a different interpretation of the ROF model, that is, the denoised image is termed as structure and the residue is treated as texture. Thus, in our work, the output of the ROF algorithm is efficiently used for keypoint detection, and also for obtaining grayscale structure and texture cues.

For combining features from different cues, a natural choice for the classification framework is the bag-of-words model, which has become the standard image classification pipeline due to its simplicity and high performance on various datasets [39, 43, 44]. Each set of local descriptors extracted for a particular cue are quantized using a visual vocabulary that is built using K-means. Then, the histogram representation for all the cues are concatenated to form the final representation. Apart from the proposed feature extraction method, we also address the issue of choosing the optimal codewords in the visual codebook, which aims to reduce the codebook size and simultaneously improve classification performance.

The dictionary used for vector quantization usually consists of several codewords that are both unnecessary and detrimental to the classification performance. Hence, many works have aimed to optimize the visual dictionary by merging codewords [48, 83] or choosing the best codebook based on global codebook measures [84, 85]. Some works also consider pruning a very large codebook using criteria like likelihood ratio [86], entropy-based minimum description

length [87], etc. Similar to [87], we select the codewords from a clustering evaluation perspective by discarding clusters with a very high entropy. In particular, high entropy clusters have members from almost all object categories, and therefore, they are potentially confusing when creating the histogram representation. However, moderately high entropy clusters may still be useful for classification, in case of shared features between different categories. To balance these two ideals, we use cross-validation to determine the usefulness of a cluster, and thus achieve reduction in codebook size and also performance boost.

To summarize, our focus in this dissertation is to investigate how to develop a robust object categorization framework that efficiently combines appearance and shape cues using the bag-of-words model.

1.3 Color Image Classification

Physiological and clinical studies in humans suggest that visual information processing is highly parallelized, and different cues such as color, depth, form, are perceived by separate channels [88]. In computer vision, this model of visual information processing has been widely adopted in the saliency detection literature (refer to [89, 90]). Drawing inspiration from the success of the saliency models, we propose parallelized local encoding for multiple object cues like color, structure, texture, and shape, using the log-polar transform in the bag-of-words framework.

In the bag-of-words framework, the first big improvement came in the form of spatial pyramid pooling [50], which aims to capture mid-level information by dividing the image into several smaller regions and encoding regional histograms

apart from the global bag-of-words histogram representation. Inspired by the spatial pyramid approach, some works have tried to modify the rigid rectangular grid pooling to obtain compact and adaptive representations [51,54]. Besides pooling techniques, much of the effort by the computer vision community has been in the direction of advanced encoding methods, which assign each local descriptor to multiple codewords instead of assigning it to the closest one (vector quantization) or extract covariance measures. Some successful techniques are sparse coding [52], locality constrained linear coding [91], Fisher vector encoding [92], vector of locally aggregated descriptors [93], radial basis coding [94], etc. The premise of all these methods is that information is lost when a local descriptor is simply assigned to the nearest codeword. While improvements have been reported for these advanced encoding methods over vector quantization, high performance improvements have been elusive under controlled conditions [95]. Moreover, the computational cost is rather high for these methods, as noted in [95]. In stark contrast to the above works, we believe that if the features are powerful enough, vector quantization's performance can be significantly higher than the reported results using various local descriptors such as SIFT, PHOW, self-similarity image descriptor, etc. To this end, we propose a multi-cue object representation using vector quantization that has significant performance improvement over several encoding methods.

Apart from advanced encoding techniques to improve the standard histogram representation, a few works have tried to capture the local statistics of the image by including contextual information, such as bi-grams [84], or Markov stationary features [57], or higher order spatial co-occurrence statistics [96]. In

general, large codebooks are associated with better performance in the bag-of-words framework, but obtaining co-occurrence statistics from large codebooks would be very noisy. For instance, a nominal codebook size of 2000 used in many works would lead to a spatial co-occurrence matrix of dimension 2000 by 2000, which is already prohibitive to store and process. Inspired by the performance of small visual codebooks in techniques like Fisher vector encoding and vector of locally aggregated descriptors, we propose to encode co-occurrence statistics using codebooks with just over hundred codewords. Thus, we can obtain an improvement in the classification accuracy while retaining computational efficiency.

Recent works using the bag-of-features model for image classification opt for a simple dense sampling strategy instead of a keypoint detector to choose the sampling locations in the image [55, 95]. Usually, the number of sampling points chosen by keypoint detectors is drastically outnumbered by the dense keypoints strategy. Consequently, keypoint detectors are often at a disadvantage, as the classification accuracy relies heavily on the number of local descriptors extracted from the image [55]. Although the simplicity of the dense grid keypoints is appealing, the computational cost of obtaining the local descriptors is very high. Therefore, we investigate the possibility of a keypoint detection scheme that produces on par performance with the dense keypoint strategy, at a much lower computational cost in terms of both memory and computational time requirements.

Similar to the strategy adopted for the grayscale images, we define keypoints as visually salient locations in the image. A salient region refers to an area that "stands-out" from its neighborhood and therefore pre-attentively captures atten-

tion. In the saliency literature, entropy has been used as a quantifying measure by many works [97–101]. However, entropy based salient detectors like AIM [99] have much lower precision and recall compared to other algorithms developed for salient region detection [90, 102] on various benchmark datasets. This implies that both the quality and quantity of pixels chosen as salient locations are sub-optimal for entropy based saliency detectors. Therefore, motivated by the lack of success for the mathematically sound information theoretic measure, we propose the continuous domain version of entropy - differential entropy - to have better discriminative power over discrete entropy for the problem of choosing keypoints for feature extraction.

In a nutshell, our focus in this thesis is to investigate and provide solutions to efficiently combine color, appearance and shape cues using the bag-of-words model, with very high performance without using any advanced encoding methods.

1.4 Application to Video Processing

Object tracking is one of the central tasks in video processing with various practical applications such as human-computer interaction [103, 104], video surveillance [105–107], vehicle navigation [108], traffic monitoring [109–111], and motion analysis [112]. In this dissertation, the focus is on traffic monitoring in expressways for automatic overspeed vehicle detection.

Over the past few years, many computer vision based traffic monitoring systems have been developed [113–119]. These methods either use a single camera or a stereo camera for monitoring the vehicles. As a result, the performance of

the above-mentioned systems is limited by the fixed depth-of-field of the cameras (small tracking range) [120,121], and they are better suited for traffic monitoring situations such as congestion control and intersection monitoring. In practice, a long tracking range is crucial when high-speed vehicle monitoring is required.

On the other hand, traditional traffic monitoring approaches using sensors such as LIDAR/RADAR have a different set of drawbacks. These approaches generally work as follows. The sensors detect the presence of a possible overspeeding vehicle and trigger a camera to capture its image. However, when many vehicles are present in the vicinity of the overspeeding vehicle, the detector will not be able to single it out, as seen in some of the wrong speed tickets issued worldwide [122, 123]. Furthermore, interference caused by big vehicles can lead to unreliable results for speed detection. Essentially, the communication gap between the sensor and the visual data limits law enforcement. Therefore, a vision-based traffic monitoring system that can address the above issues without succumbing to the problem of a small tracking range is desirable.

Reference [124] proposed a multiple depth-of-field image sensor, inspired by the vision of raptors, for deep-field object tracking. Compared to traditional cameras, [124] used the composite image information to detect the presence of vehicles over a longer range. However, only a limited portion of the tracking result was used as a switch to activate another camera for license plate detection. In fact, this can be achieved by a single camera or other sensors. In other words, [124] only showed that tracking could be done in the deep field without utilizing the full tracking range for a suitable application. Moreover, the system developed in [124] is capable of tracking only one object at a given instant. In

contrast, we consider tracking multiple objects simultaneously by making use of USB 3.0 cameras instead of the surveillance cameras used by the authors of [124]. It is to be noted that the usage of USB 3.0 cameras is crucially important for the quality of speed detection, because they offer high frame rates with minimal frame dropping and also support synchronous video acquisition with easy plug-and-play ability with laptops. On the contrary, surveillance camera standards such as Gigabit Ethernet/Fire Wire/Camera Link require a special communication device for data acquisition and a separate power supply. Additionally, the high bandwidth and the portability of USB 3.0 cameras offer a great alternative to the cumbersome and unreliable (high frame dropping) cameras used in [124].

In this thesis, we propose a speed detection application that utilizes the full tracking range (up to 1 km) acquired from cameras of different visual field depths. To this end, we utilize the log-polar transform to stitch the visual information obtained by each camera into a single video stream, and track the vehicles in the LPT space instead of the Cartesian space. Then, the tracking information is used to trigger a third camera for capturing the license plate information whenever a vehicle is detected to be exceeding the allowed speed threshold.

In summary, we aim to develop a composite vision system with multiple depth-of-field viewing ability that extends the tracking range of traditional traffic monitoring systems.

1.5 Objectives and Contributions

As discussed in the previous sections, despite the extensive work in the field of object classification and video processing, there are still challenges that have

not been addressed thoroughly. The principal aim of this thesis is to develop a novel object classification framework that incorporates object cues using invariant local features and contextual information, and extend the tracking abilities of current vision-based traffic monitoring systems by including multiple depth-of-field information. The main contributions of our work are as follows.

1. **Binary Shape Classification**: We propose a classification framework for binary shapes that have scale, rotation and strong viewpoint variations. To this end, we develop several novel techniques. First, we employ the spectral magnitude of log-polar transform as a local feature in the bag-of-words model. Second, we incorporate contextual information in the bag-of-words model using a novel method to extract bi-grams from the spatial co-occurrence matrix. Third, a novel metric termed 'weighted gain ratio' is proposed to select a suitable codebook size in the bag-of-words model. The proposed metric is generic, and hence it can be used for any clustering quality evaluation task. Fourth, a joint learning framework is proposed to learn features in a data-driven manner, and thus avoid manual fine-tuning of the model parameters. We test our shape classification system on the animal shapes dataset and significantly outperform state-of-the-art methods in the literature.

2. **Grayscale Image Classification**: We propose a cue-based object categorization framework to extract different types of image information, and fuse it to obtain better discriminative power. Specifically, we used the Rudin-Osher-Fatemi method to decompose the grayscale image into the structure and texture parts, and extracted local features using the log-

polar transform. Furthermore, local shape descriptors are extracted using a state-of-the-art salient object detection model to account for contour cues. The extracted local descriptors are quantized using the bag-of-words representation with some key contributions: (1) a keypoint detection scheme based on image denoising is proposed to select sampling locations, and (2) a codebook optimization scheme based on discrete entropy is proposed to reduce the number of codewords and at the same time increase the overall performance. We tested our framework on the ETH-80 dataset using the leave-one-object-out cross validation method and obtained a very high improvement in classification performance compared to state-of-the-art methods.

3. **Color Image Classification**: We propose a multi-cue object representation for color image classification using the standard bag-of-words model. Ever since the success of the bag-of-words model for image classification, several modifications of it have been proposed in the literature. These variants target to improve key aspects, such as efficient and compact dictionary learning, advanced image encoding techniques, pooling methods, and efficient kernels for the final classification step. In particular, "soft-encoding" methods such as sparse coding, locality constrained linear coding, Fisher vector encoding, have received great attention in the literature, to improve upon the "hard-assignment" by vector quantization. However, these methods come at a higher computational cost while little attention has been paid to the extracted local features. In contrast, we propose a novel multi-cue object representation for image classification using the

simple vector quantization, and show highly competitive classification performance compared to state-of-the-art methods on popular datasets like Caltech-101 and MICC Flickr-101. Apart from the object representation, we also propose a novel keypoint detection scheme that helps to achieve a classification rate comparable to the popular dense keypoint sampling strategy, at a much lower computational cost.

4. **Application to Video Processing**: We present a novel vision-based traffic monitoring system, which is inspired by the visual structure found in raptors, for tracking expressway vehicles and estimation of their real-world speed. This vision system also features a license plate detection camera which is triggered whenever there is an instance of overspeeding. One of the main novelties of the proposed system is the usage of multiple depth-of-field information in log-polar space for tracking expressway vehicles over a longer range compared to the typical Doppler effect-based RADAR or LIDAR traffic monitoring systems. Thus, the proposed system provides accurate speed information for overspeed vehicle detection using computer vision techniques. To this end, a novel speed calculation algorithm is proposed for the composite vision information acquired by the system, and with the aid of a license plate detection camera, identity information of the overspeeding vehicles can be recorded for law enforcement. The calculated speed was verified using RADAR speed detection systems and smartphone applications, and the deviation was found to be within ±3 km/hr compared to the real-world driving speed. In summary, the proposed system provides a novel solution to improve the capabilities of traffic monitoring using vision

based methods.

1.6 Thesis Organization

The remainder of the thesis is organized as follows.

Chapter 2 introduces the log-polar sampling technique used in the subsequent chapters of the thesis. Preliminary evaluations are given to test the ability of log-polar transform to classify binary shapes with scale and rotation variations using a simple global classification framework.

Chapter 3 presents a local feature based binary shape classification framework for dealing with scale, rotation, and strong view-point variations. Extensive evaluations of the framework on a challenging benchmark database is given.

Chapter 4 further introduces a grayscale image classification framework that efficiently combines appearance and shape cues. Apart from the comparison to state-of-the-art solutions, comparative evaluations of the performance of the appearance cues and the shape cue are made on a popular benchmark dataset.

Chapter 5 presents a generic image classification framework for color images that efficiently combines color, appearance and shape cues. Extensive evaluations on two standard object datasets are reported.

Chapter 6 introduces a video processing application that is based on the log-polar sampling technique extensively used in this thesis. The verification of the speed output of the proposed traffic monitoring system is done using RADAR speed detection systems and smartphone applications.

Chapter 7 presents our conclusions and indicates future research directions.

Chapter 2

Scale and Rotation Invariance using Log-Polar Transform

2.1 Introduction

As mentioned in Chapter 1, heuristically derived local descriptors like SIFT do not guarantee scale and rotation invariance from a theoretical point of view. Recently, an attempt [125] was made to theoretically explain the phenomenal success of the SIFT descriptor. This study has proven that SIFT is scale and rotation invariant under certain conditions. However, scale and rotation invariance is achieved only for the selected keypoints using the scale-space and does not apply to other useful structures, like edges in the image [36]. Moreover, the common method of using SIFT descriptor without keypoint selection in object classification systems, though widely reported to give good performance, is obviously without the guarantee of the invariant properties.

Scale and rotation changes in the Cartesian image correspond to horizontal

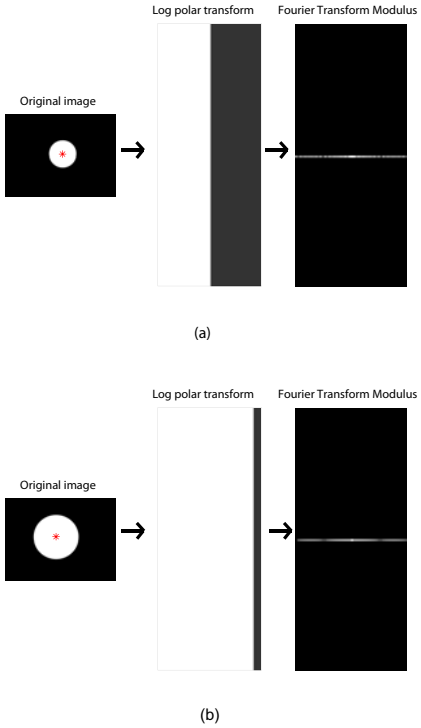
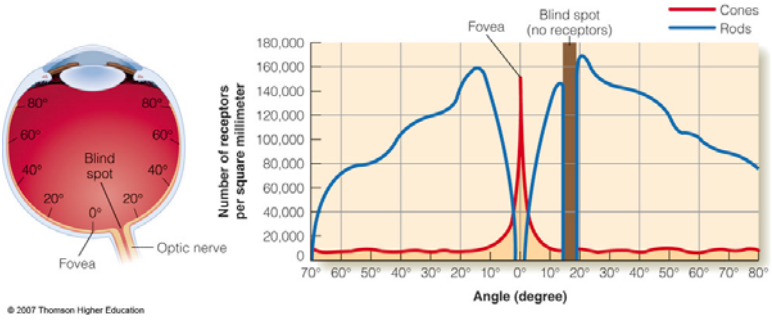
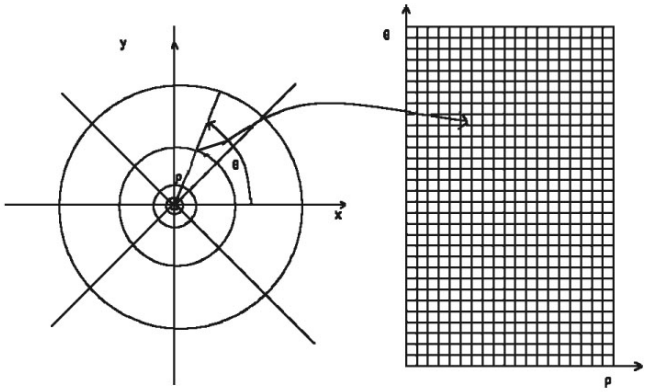


Figure 2.1: Log-polar transform applied to the image by centering on the shape, followed by computing the Fourier transform modulus. Scale change in the Cartesian space corresponds to a horizontal shift in the log-polar space, which can be eliminated by computing the Fourier transform modulus to obtain a scale invariant descriptor for each binary shape.

and vertical shifts in the log-polar domain, respectively [32]. Note that the Fourier transform modulus (magnitude of the Fourier transform) of two images related by pure translation is the same. Consequently, two log-polar images of similar shapes, which have scale and rotation variations, are expected to have "similar" Fourier transform magnitude. This concept is illustrated in Figure 2.1 for the simple case of a circle to facilitate easy visual comparison in the frequency domain. After eliminating the translation differences in the log-polar space, it is easy to see that circles of different radii have nearly identical features in the frequency domain. Therefore, our choice of LPT is motivated by this sound mathematical foundation for ensuring scale and rotation invariance, which



(a) Primates' retina photoreceptor distribution.



(b) Mapping from Cartesian (x, y) to log-polar space (ρ, θ).

Figure 2.2: Biologically inspired log-polar mapping.

heuristically designed feature descriptors fail to achieve.

The remainder of this chapter provides details about the log-polar sampling technique along with preliminary evaluation results on several shape databases.

2.2 Log-Polar Transform

By observing the non-uniform distribution of cones in the primate fovea, as shown in Figure 2.2(a), a logarithmic relationship for information around the fovea structure can be established [126]. Therefore, by considering an exponential sampling of the Cartesian image, the log-polar transform simulates the foveal mechanism of the human vision system. In other words, there is dense sampling

near the center of the log-polar grid and coarse sampling towards the periphery (see Figure 2.2(b)).

Let us define the mapping from Cartesian coordinates of the image - (x, y) to LPT coordinates - (ρ, θ) as follows,

$$x' = r\cos\theta, \qquad y' = r\sin\theta, \tag{2.1}$$

where (r, θ) are polar coordinates defined with (x_c, y_c) as the center of the transform and $(x', y') = (x - x_c, y - y_c)$, that is,

$$r = \sqrt{(x')^2 + (y')^2}. \tag{2.2}$$

The angle θ is required to be in the range $[0, 2\pi)$, but arctan is defined only for $(\frac{-\pi}{2}, \frac{\pi}{2})$. Therefore, the angles are computed depending on the quadrant as shown below.

$$\theta = \begin{cases} \arctan\left(\frac{y'}{x'}\right) & \text{if } x' > 0 \\ \arctan\left(\frac{y'}{x'}\right) + \pi & \text{if } x' < 0 \\ +\frac{\pi}{2} & \text{if } y' > 0, x' = 0 \\ +\frac{3\pi}{2} & \text{if } y' < 0, x' = 0 \\ \text{undefined} & \text{if } y' = 0, x' = 0 \end{cases} \tag{2.3}$$

The above operation produces output in the range $(\frac{-\pi}{2}, \frac{3\pi}{2}]$, which can be mapped to $[0, 2\pi)$ by adding 2π to negative values. The convention of the log-polar parameters in [35] has been adopted here: (1) The radii of the smallest is represented as r_{min}. (2) The maximum radius is represented as r_{max}. (3) The logarithmic scal-

ing is defined as $\rho = \log r$. (4) The number of rings and wedges are represented as n_r and n_w, respectively.

The samples of LPT lie at the intersection between rings and wedges, and thus the size of the log-polar image is n_r by n_w. In general, the intersection happens at arbitrary locations in the image, and therefore bilinear interpolation is used to find the image intensity at these locations. Bilinear interpolation considers the closest 2×2 neighborhood of known pixel values surrounding the unknown value. For instance, if (x, y) is the location of the unknown value in an image I, and (x_1, y_1),(x_1, y_2), (x_2, y_1) and (x_2, y_2) are the surrounding pixel locations, then the image intensity $I(x, y)$ is given by a weighted summation,

$$\begin{aligned} I(x,y) = &\frac{(x_2 - x)(y_2 - y)}{(x_2 - x_1)(y_2 - y_1)} I(x_1, y_1) + \frac{(x - x_1)(y_2 - y)}{(x_2 - x_1)(y_2 - y_1)} I(x_2, y_1) \\ &+ \frac{(x_2 - x)(y - y_1)}{(x_2 - x_1)(y_2 - y_1)} I(x_1, y_2) + \frac{(x - x_1)(y - y_1)}{(x_2 - x_1)(y_2 - y_1)} I(x_2, y_2). \end{aligned} \tag{2.4}$$

Due to this non-equidistant polar sampling, scale and rotation changes in the Cartesian image correspond to horizontal and vertical shifts in the log-polar domain, respectively. Nevertheless, log-polar transform when used as a global descriptor, as shown in Fig. 2.1, is sensitive to changes in the location of its centroid on the image, i.e., greater the center mismatch between two shapes, greater the image distortion [32]. In other words, noise and occlusion would severely affect the invariant properties of LPT, which is the principal drawback of using it as a shape descriptor.

In the computer vision literature, the most successful application of log-polar mapping has been the shape context [29]. However, it is to be noted that shape context creates log-polar histograms instead of using the original

LPT, which is sampling the image at the intersection of rings and wedges of the transform. Therefore, when using the original LPT as a centroid-based global shape descriptor, additional feature extraction would be required to decrease the effects of noise and misalignment due to occlusions and shape distortions. In the next section, we present a global shape classification framework using LPT to test its effectiveness in dealing with scale and rotation changes under noisy scenarios.

2.3 Global Shape Classification using LPT

For each training image, log-polar sampling is done by centering on the shape's centroid, followed by the discrete Fourier transform to obtain a scale and rotation invariant descriptor. Using all the training descriptors, the feature extraction module finds a discriminant low-dimensional subspace. After projecting the training descriptors to the subspace, the new shape descriptors are simply stored to be used in the testing stage. For a test image, the Fourier transform modulus of LPT is projected to the low-dimensional subspace and compared with the training descriptors for classification using the nearest-neighbor method. In the following subsection, the feature extraction module is described in detail.

2.3.1 Feature Extraction

Feature extraction is meant to improve the performance of the classifier, by discarding irrelevant information such as noise and redundancy from the set of input features [127]. While noise can be readily regarded as a hindrance to optimal classification, it has also been observed that if the number of training

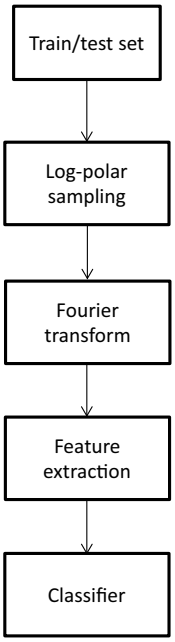


Figure 2.3: Flowchart of the global shape classification system based on LPT.

samples is far less than the feature dimension (small sample size), then the 'curse of dimensionality' degrades the performance of the classifier [128]. As a global descriptor, the log-polar sampling produces an order of 10^4 features, which would increase the computational complexity of the classifier and affect its performance on the test data. Although extra computational effort is required for discriminant feature analysis in the training stage, the classifier performance can be speeded-up after projecting to the low-dimensional subspace. With this simple projection step, the classifier can be made robust to noise while maintaining real-time performance. Moreover, discriminant analysis is essential for differentiating similar shape categories studied in this thesis.

Among the many feature extraction methods in the literature, three standard techniques, namely principal component analysis (PCA) [129], Fisher's linear discriminant (FLD) [130] and recursive FLD (RFLD) [131] are explored to improve the shape classification accuracy and reduce computational load during

the testing phase. A brief overview of the feature extraction techniques is given below.

Principal Component Analysis

PCA is an unsupervised linear feature extraction method that is largely exploited for dimensionality reduction. For a set of N d-dimensional samples $(x_1, x_2, \cdots, x_N)$ with N_i samples in the subset D_i belonging to class ω_i, $(i = 1, \cdots, C)$, PCA seeks a projection W that minimizes the error function:

$$J_{PCA}(W) = \sum_{k=1}^{N} ||x_k - y_k||^2 \tag{2.5}$$

where y_k is obtained after projection of x_k by W as $y_k = WW^T x_k$. The minimization is equivalent to finding the eigenvectors of the total scatter matrix, defined as:

$$S_T = \sum_{k=1}^{N} (x_k - \mu)(x_k - \mu)^T \tag{2.6}$$

where μ is the mean of all training samples:

$$\mu = \frac{1}{N} \sum_{k=1}^{N} x_k. \tag{2.7}$$

The columns of W associated with non-trivial eigenvalues are called the principal components (PCs), and those with negligible eigenvalues are regarded as arising from noise.

Fisher's Linear Discriminant

FLD is a supervised feature extraction method to minimize the within-class and between-class scatter, i.e., it maximizes the following objective function,

$$J_{FLD}(w) = \frac{w^T S_B w}{w^T S_W w}. \tag{2.8}$$

The between-class scatter matrix S_B is defined as follows:

$$S_B = \sum_{i=1}^{C} N_i(\mu_i - \mu)(\mu_i - \mu)^T \tag{2.9}$$

where N_i is the number of samples in each class and μ_i is the sample mean of class i. The within-class scatter matrix S_W is given by,

$$S_W = \sum_{i=1}^{C} S_i \qquad \text{where } S_i = \sum_{x \in D_i} (x - \mu_i)(x - \mu_i)^T. \tag{2.10}$$

The vector w that maximizes equation (2.8) must satisfy:

$$S_B w = \lambda S_W w. \tag{2.11}$$

If S_W is full-rank, we can obtain a conventional eigenvalue problem by obtaining the inverse of S_W as $S_W^{-1} S_B w = \lambda w$. However, often due to the limited number of training samples compared to the dimension of the samples, S_W is singular. Typically, PCA is employed to reduce the feature dimension and make S_W non-singular. It can be seen that FLD returns utmost $C-1$ features because the rank of S_B is utmost $C - 1$. To overcome this limitation, recursive Fisher's linear discriminant can be used to extract more than $C - 1$ features [131]. The

main idea behind RFLD is to recursively perform FLD while ensuring that the extracted features are orthogonal to each other. Due to space constraints, the reader is directed to [131] for the technical details of RFLD.

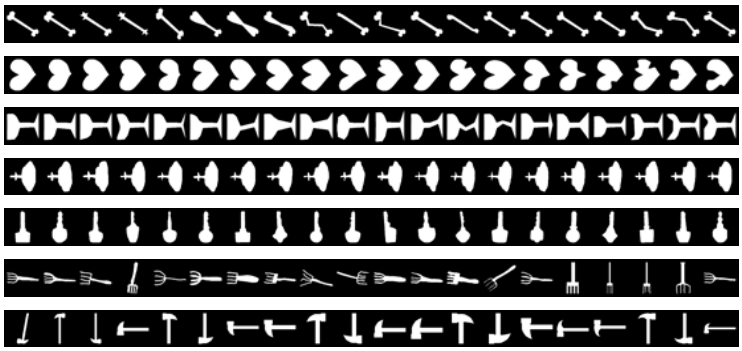
2.4 Experimental Results and Discussion

The global shape classification framework described in the earlier section was tested on five publicly available shape datasets. The selection of the databases were done keeping in mind to test our framework for similar & noisy binary shapes. We selected three datasets following the work of [132]. The authors of [132] tested their algorithm on shapes created from automatic segmentation methods that result in noisy artifacts at the shape boundaries. Furthermore, they selected shape categories that exhibit high inter-class similarities. We also selected two widely used benchmark datasets, Kimia-216 [133] and Chicken pieces silhouettes database [134], to test our shape classification framework. Figure 2.4 shows sample shapes from the five databases.

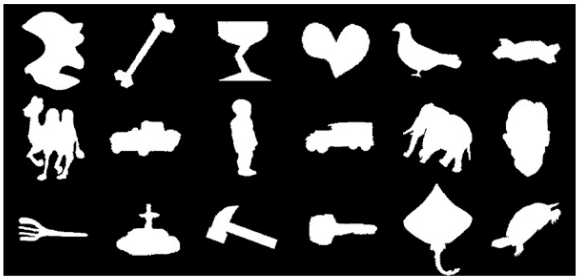
Table 2.1 & 2.2 shows the classification accuracy of the proposed framework in comparison to previous works in the literature. We outperform the state-of-the-art methods on three datasets and perform on par with state-of-the-art algorithms on the Kimia-216 and the MPEG-7 dataset. From the results of MPEG-7, Kimia-216 and the Chicken pieces datasets, it is clear that feature extraction plays a crucial role in obtaining high performance when dealing with similar and noisy shape categories. Although PCA helps to reduce the computational load during the classification stage, its accuracy is the same as that of the direct nearest-neighbor method without feature extraction, because we consider

(a) Airplane shapes – 7 categories.

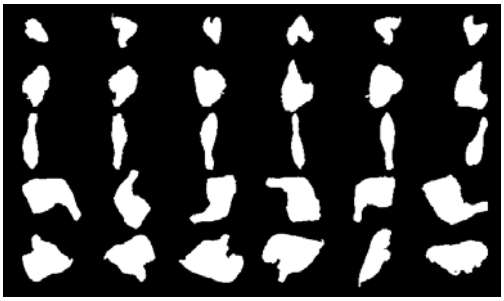
(b) Vehicle shapes – 4 categories column-wise.



(c) Subset of MPEG-7 – 7 categories row-wise.



(d) Kimia216 – 18 categories.



(e) Chicken shapes – 5 categories row-wise.

Figure 2.4: Shape databases used in this work.

Table 2.1: Comparison of shape classification accuracies on three publicly available shape databases (%).

	Airplanes (7 classes)	Vehicles (4 classes)	MPEG-7 (7 classes)
Ref. [132]	99.05	84.17	96.43
Ref. [135]	99.00	87.00	–
Ref. [136]	99.42	–	–
Ref. [137]	–	–	**98.80**
Ref. [138]	–	85.42	–
1-NN	100	98.33	92.86
PCA	100	98.33	92.86
FLD	**100**	98.33	97.14
RFLD	**100**	**99.17**	98.57

all the principal components and do not alter the distribution of the samples. In contrast, Fisher's linear discriminant and its variant, recursive FLD, show high capability in rejecting noisy features and overcoming the small sample size problem. In general, RFLD slightly outperforms FLD due to the flexibility of the number of features that can be extracted from the input data. Note that we used a simple nearest-neighbor classifier to demonstrate the effectiveness of the features. In other words, it is possible that even if the extracted features are not discriminatory, high performance can be achieved using powerful classifiers like neural nets or SVM [127].

2.4.1 Discussion

The airplane shapes database (Fig. 2.4(a)) has seven classes with each class having 30 samples, making a total of 210 shapes. Tenfold cross validation was carried out following the protocol of [132] and [135]. First, the dataset was split into 10 non-overlapping sets of (almost) equal size while maintaining the class balance in each split. Then, we combined 9 of these for training and the remaining one was used for testing. This process was repeated for the other

Table 2.2: Comparison of shape classification accuracies on the Kimia-216 and the Chicken shape database (%).

	Kimia-216 (18 classes)	Chicken (5 classes)
Ref. [139]	94.1	–
Ref. [140]	97.2	–
Ref. [141]	**97.7**	–
Ref. [142]	95.4	86.48
Ref. [143]	–	87.16
Ref. [144]	–	84.45
Ref. [145]	–	81.10
1-NN	94.44	71.14
PCA	94.44	71.14
FLD	**97.7**	85.91
RFLD	**97.7**	**87.91**

combinations of training and testing sets. Unlike [132], we neither filtered the shapes to reduce the effect of noise nor normalized the shape perimeter to deal with scale changes. On this well-segmented dataset, we could achieve perfect classification without requiring any feature extraction methods, just by exploiting the invariant properties of log-polar transform.

In the second set of shape classification experiments, vehicle shapes extracted from traffic videos are to be classified into one of four classes: sedan, pickup, minivan and SUV. The shapes are distorted in the bottom half due to shadows, and no pre-processing was done to remove them (Fig. 2.4(b)). Each class has 30 samples, making a total of 120 shapes. Tenfold cross validation was carried out following the protocol of [132], wherein we obtained a huge improvement in the classification results compared to previous works (from 85% to 99%). Note that simply using the nearest-neighbor classifier without feature extraction already yields 98% classification accuracy, which suggests that uniform boundary noise can be handled well by the invariant properties of log-polar transform. Fur-

thermore, improvements in classification accuracy can be achieved using feature extraction.

The MPEG-7 database has 1400 shapes with 20 samples for each class. A subset of this dataset, containing 7 shape classes with 140 samples in total, was used in [132, 137]. Table 2.1 shows the classification accuracy of the proposed method, compared to the best results obtained in [132]. Clearly, without feature extraction, the LPT features perform poorly compared to the benchmark accuracy. This is due to the strong distortions, occlusion and inter-class similarities of the samples, especially among key, bone, and hammer categories. With feature extraction, the classification accuracy can be improved to 98.6%, which is comparable to the accuracy reported in [137].

Kimia-216 [133] (Fig. 2.4(d)) is a larger subset of MPEG-7, containing 18 shape categories with 12 shapes per category. Following previous works, we carried out leave-one-out cross validation to obtain the overall classification accuracy. In comparison to more complex algorithms in the literature, the proposed method outperforms all except one (Table 2.2). Again, the classification accuracy is lower without discriminant feature extraction while PCA does not improve upon the direct 1-NN. Both RFLD and FLD analysis produce the same classification accuracy, which is also equal to the best accuracy of reference [141].

The chicken pieces dataset consists of 446 shapes of five different chicken parts, namely: wing, breast, leg, thigh and quarter (Fig. 2.4(e)). In comparison to the other four datasets considered so far, the shapes in this database have strong view-point variations. Following the protocol in the literature, we randomly divided this dataset into three subsets: 149 shapes for training, 149

shapes for validation and the remaining 148 shapes for testing [142]. Although we outperform all the methods in the literature using RFLD analysis, the performance is not significantly different from that of [143]. In the next subsection, we study the effects of varying the LPT parameters on the classification accuracy.

2.4.2 Effect of Varying LPT Parameters

The blind spot for the log-polar sampling is decided by the minimum radius r_{min}, inside which sampling is not performed. We simply set the minimum radius to be 1 pixel to extract as much information as possible. The maximum radius r_{max} is set as the maximum radius of the shape image. To study the effect of varying the LPT grid resolution, we randomly chose several settings for which $n_r > 50$ and $n_w > \frac{360}{6}$, and compared them in terms of classification accuracy. The results of the comparison study are tabulated in Table 2.3 for the Kimia-216 and Chicken pieces datasets, with FLD as the feature extraction method. It is evident that simply increasing the number of rings and wedges does not guarantee higher classification accuracy, which is especially true for the Chicken pieces database. The highest resolution considered was 200 rings and 360 wedges, which failed to give the best classification rate on both the databases. However, the results are quite stable for many choices of the parameters.

Besides the LPT grid resolution, the minimum radius r_{min} was chosen to be 1 pixel without any estimation. Fig. 2.5 shows the effect of varying the minimum radius on the classification accuracy. For the Vehicles and MPEG-7 datasets, $r_{min} = 3$ gives slightly better accuracy compared to $r_{min} = 1$. However, the comparison studies in Fig. 2.5 are not entirely conclusive, since $r_{min} = 1$ gives the highest accuracy for the Chicken pieces dataset. In general, we

Table 2.3: Comparison of classification accuracy using various LPT grid resolutions (%).

Grid resolution (n_r by n_w)	Kimia-216	Chicken
120×90	97.70	**85.91**
120×360	97.70	84.56
120×180	97.70	83.22
130×90	95.83	83.22
150×90	97.22	82.55
180×90	97.70	81.21
90×360	**98.15**	**85.91**
200×360	97.22	85.23
190×180	**98.15**	**85.91**
200×90	97.70	85.23

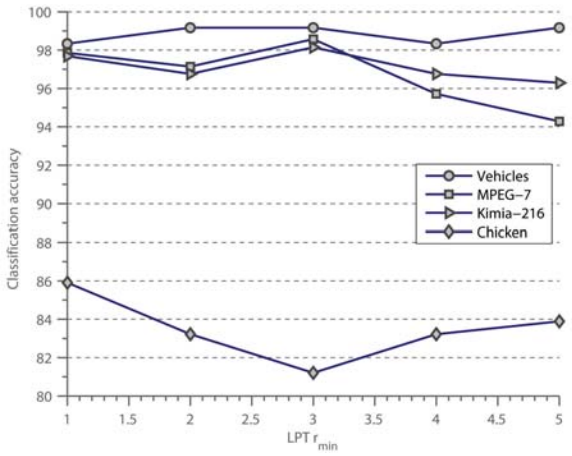


Figure 2.5: LPT minimum radius vs. classification accuracy.

conclude that a minimum radius of 1 pixel is one of the best choices in terms of classification accuracy. Besides pure accuracy value evaluations, we demonstrate the robustness of the LPT shape descriptor using a self-organizing map [146] (SOM).

2.4.3 Robustness Analysis using Self-Organizing Map

By adopting a strategy originally proposed by Kohonen [146], the SOM establishes complex relationships that exist among high-dimensional input patterns into a two-dimensional pattern. Figure 2.6 shows the structure of a SOM. Each

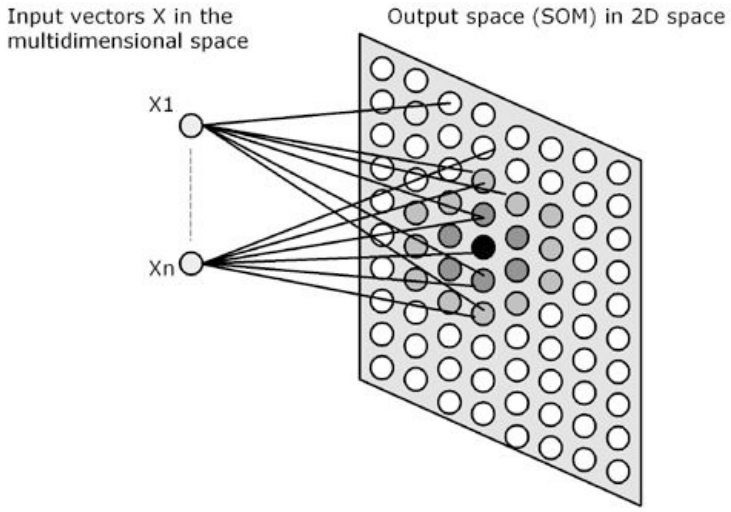


Figure 2.6: Structure of the self-organizing map.

neuron is represented by a d-dimensional weight-vector W_i, where d is the dimension of the LPT feature vector. Neurons are connected to adjacent neurons by a neighborhood relation that characterizes the topology of the network. The network is trained iteratively as follows:

1. Randomly select one sample vector from the input data set, and calculate the Euclidean distances between it and all the weight-vectors W_i of the network.

2. Find the best matching unit, whose weight-vector is closest to the input vector. Call this neuron, c.

3. Update the weight-vectors of the network, such that the best matching unit is moved closer to the input vector.

4. Go to steps 2 and 3, repeat until there are no significant changes while updating weight-vectors.

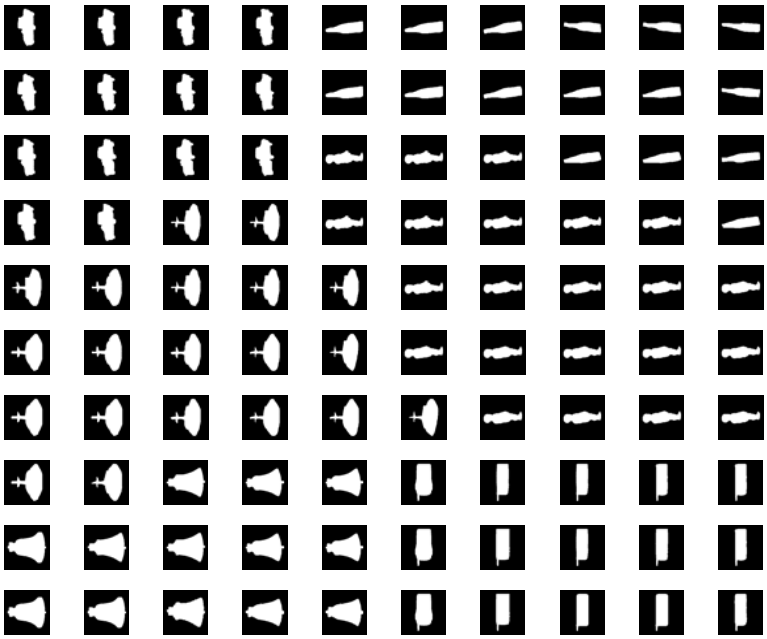


Figure 2.7: Nearest neighbors of the SOM neurons. The weight vector of each neuron is matched to the closest shape descriptor and the corresponding shape image is displayed.

In our analysis, we represent all the shapes in the MPEG-7 subset using log-polar transform followed by its Fourier transform modulus, and feed the FFT features to a self-organizing neural network of size 10 by 10. After training, the SOM exhibits distinct clusters for various shapes (Fig. 2.7), thereby demonstrating that the LPT features have the property of scale and rotation invariance, and also, are meaningful features for shape classification under noisy conditions.

2.4.4 Computation Time

In this age of information explosion, one of the most important applications of shape classification is content-based image retrieval (CBIR) [147]. Most CBIR systems store image content as visual features belonging to one of four categories,

namely color, texture, shape and structure [148, 149]. Therefore, modern-day shape classification systems are required to possess an accurate as well as efficient methodology. Besides high accuracy achieved by the proposed method on several benchmark datasets, the computational time is also very low compared to many recently proposed shape classification algorithms. It is hard to compare the various methods in the literature in terms of computation time, mainly due to the lack of publicly available code. While most papers choose not to report the computation time of the classification step, we rely on a handful of papers for comparison.

The main bottleneck for the LPT shape descriptor comes from the computation of the 2-D FFT. If the number of pixels in the log-polar image is N, the computational complexity is $O(N * log(N))$, which is orders of magnitude less than that of ref. [144], whose shape context based computation complexity is $O(N^2)$. Next, we directly compare the processing time of our classification algorithm with those reported in the literature.

The classification time reported by ref. [139] for the Kimia-216 dataset was 25 min and ref. [140] reported to take 45 min. In comparison to these methods, the proposed shape classification framework took only 3 min (includes training time) to classify all 216 shapes of the Kimia-216 dataset. This is substantially lower than the methods in the literature while not compromising on classification accuracy. For the Chicken pieces dataset, ref. [144] reported that computing the pair-wise similarity between two shapes took 76.5 ms on an average. On the other hand, it takes less than 150 ms for the whole testing stage using the proposed shape classification framework, i.e, reading the image, log-polar

Table 2.4: Computation time of the proposed shape classification system for classifying a test image (in ms).

	Testing time
Airplanes	138
Vehicles	132
MPEG-7	133
Kimia-216	143
Chicken	137

sampling, Fourier transform, feature extraction and classification (Fig. 2.3). The proposed algorithms were implemented in MATLAB on an Intel Core i7-2600 CPU @ 3.4 GHz with 8 GB memory. Since we did not optimize the code for speed using a C/C++ implementation, there is still room for improvement in terms of computation time. The computation time for the testing stage for all the databases is tabulated in Table 2.4.

2.5 Summary

We proposed a global shape classification system using a biologically inspired sampling technique called log-polar transform, which achieves scale and rotation invariance by simulating the distribution of cones in the retina. The performance of the proposed shape classifier was tested on five datasets, viz.: Fighter airplanes, Vehicles, Subset of MPEG-7, Kimia-216 and Chicken pieces. The classification accuracy of the proposed method was demonstrated to be superior or on par with more complex algorithms proposed in the literature. For the airplanes and the vehicles dataset, we achieved superior performance even without feature extraction. However, for the other three datasets that has distortions, occlusion or view-point variations, feature extraction was required to close in on the performance of the previous works in the literature. This implies that the

global features extracted in the spectral domain of the log-polar transform are discriminatory as long as there is no occlusion or strong view-point variations. In other words, the centroid problem of the log-polar transform becomes more difficult to handle using global analysis, because of the non-uniformity of interior information across images of the same class. Therefore, log-polar transform as a local feature would be more suitable in such scenarios, which will be studied in the next chapter.

Chapter 3

Shape Classification using Invariant Local Features and Contextual Information

3.1 Introduction

As deduced in the earlier chapter, shapes with occlusion and strong view-point variations pose difficulty to global analysis. Therefore, this chapter aims to employ log-polar transform as a local feature for classifying binary shapes with scale, rotation, occlusion and strong view-point variations. The key idea is to sample each boundary point of the shape using log-polar transform, which is followed by computing its Fourier transform modulus. Subsequently, the scale and rotation invariant local descriptor is obtained by converting the two-dimensional Fourier transform output into a vector and performing normalization using the Euclidean norm. The extracted local descriptors are quantized using the bag-of-

words model while incorporating contextual information for improving the image representation.

The rest of this chapter is organized as follows. Section 3.2 presents the shape classification system with implementation details; Section 3.3 presents the experimental results and discussion, followed by conclusions in section 3.4.

3.2 Contextual Bag-of-Words Model

Binary shapes are classified using the bag-of-words framework consisting of four main stages: keypoint detection, feature extraction, vector quantization, and classification. In this work, keypoint detection is simply the selection of boundary points of the binary shape. Feature extraction involves sampling the binary shape at the keypoints, using log-polar transform, followed by computing its Fourier transform modulus. For the training set, the extracted descriptors are collectively used for K-means to obtain a codebook. The quantization step is the histogram representation of each training/testing image, using the codebook generated in the previous step. Then, the histograms of the training images are used to train an SVM classifier. During testing, the codebook construction step is bypassed, and a test image is simply represented using the codebook and classified using SVM. The block diagram of the proposed shape classification system is shown in Figure 3.1.

3.2.1 Feature Extraction

As seen in Chapter 2, log-polar transform is sensitive to changes in the location of its centroid on the image, i.e., greater the center mismatch between two

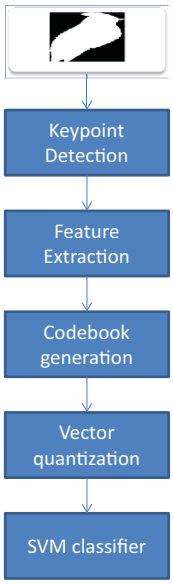


Figure 3.1: Block diagram of the shape classification system.

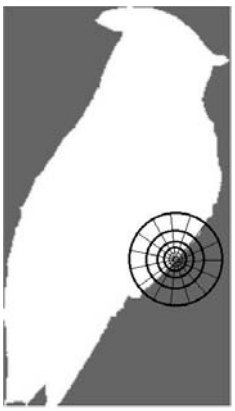


Figure 3.2: Feature extraction using log-polar transform at the shape boundaries.

shapes, greater the image distortion [32]. In other words, occlusion and viewpoint change severely affects the invariant properties of LPT. To address this issue, we choose to place the centroid of LPT at the shape boundaries and extract local features instead of a global representation, as shown in Figure 3.2. This line of reasoning is backed up by the dominance of local feature-based approaches (over global approaches) for various recognition tasks in computer vision [10]. Even so, we verify our choice of the local approach by comparing it with the global application of LPT.

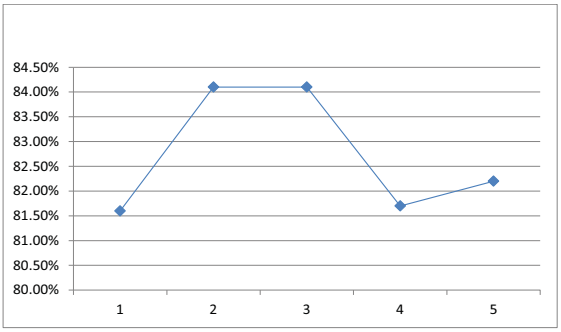
Table 3.1: Comparison of local and global approach using LPT in terms of classification accuracy (%).

	Local LPT approach	Global LPT approach
Accuracy	78.30	53.70

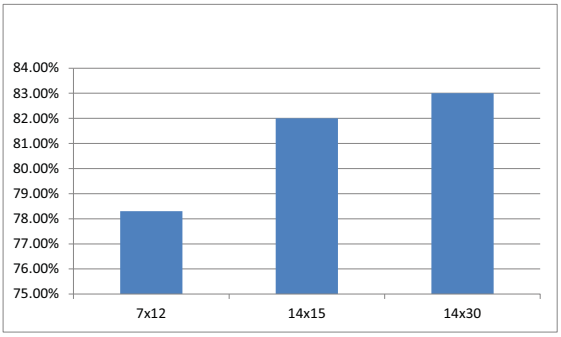
For each training/testing image, log-polar sampling is done by centering on the shape's centroid, followed by computing the Fourier transform modulus to obtain a global shape descriptor (Figure 2.1 shows this step). Then the descriptors from the training set are analyzed to find a discriminant low-dimensional subspace, using principal component analysis [150] and recursive Fisher's linear discriminant [131]. After projecting the training descriptors to the subspace, the resultant descriptors are used to train an SVM classifier. For a test image, the Fourier transform modulus of LPT is projected to the low-dimensional subspace and classified using SVM.

Table 3.1 compares the classification accuracies on the animal shapes database [151] using the global and local LPT approach. In the animal shapes dataset (Fig. 3.7), occlusion (caused by self) is a major problem, due to viewpoint variation and unavailability of interior information. Therefore, it is natural that the local approach easily outperforms the global approach, which reaffirms our choice of local LPT descriptors and the bag-of-words model. Due to space constraints, we only report the result using the best parameter settings of LPT for the global approach - minimum radius $r_{min} = 1$, maximum radius r_{max} = max. radius of shape image, number of rings $n_r = 120$, and number of wedges $n_w = 180$. The parameter settings of the proposed LPT local approach are discussed in detail below.

Intuitively, the minimum radius would not play a significant role as feature



(a)



(b)

Figure 3.3: (a) LPT minimum radius vs. classification accuracy. (b) LPT grid size vs. classification accuracy.

extraction is done for every boundary point. Reference [12] recommends 3 to 5 pixels as the minimum radius for object detection in grayscale images. However, shape context [29] obtained good results with 2 pixels for shape classification in binary images. Similarly, we found that using a minimum radius of 2 pixels is one of the best in terms of classification accuracy (Figure 3.3(a)). In the literature, shape context [29] quantized the angle into 12 divisions (n_w) and log-distance into 5 divisions (n_r). It could afford such a coarse sampling (5 x 12) due to the histogram-style treatment of log-polar transform. Nevertheless, when using the original log-polar transform, it was found that a denser sampling is required to obtain higher classification accuracies (Figure 3.3(b)). Spatial pyramid matching [50] was used to obtain the classification accuracy for the local LPT approaches shown in Fig. 3.3.

It is not straightforward to determine the maximum radius of LPT as a local feature; too small a radius will make the feature non-discriminatory and too large a radius would not result in a local feature. Reference [35] makes a reasonable suggestion to keep every pixel's orthogonal neighbors about equal distances from it, by applying the following constraint,

$$r_{max} = r_{min} \times e^{2\pi \frac{(n_r - 1)}{n_w}} \quad . \tag{3.1}$$

Using $r_{min} = 2$, $n_r = 14$, and $n_w = 30$ in equation (3.1) results in a maximum radius of about 30 pixels, which was used for illustration in Fig. 3.2. This may still be sub-optimal in terms of classification accuracy. Hence, a procedure to set the maximum radius in a data-driven manner is presented later in this section.

3.2.2 Codebook Selection

The descriptors obtained from the training images are collectively used to obtain a codebook, using VLFeat's [152] implementation of K-means with an accelerated Elkan algorithm for optimization. The codebook is simply the cluster centroids obtained using K-means. In order to evaluate the clustering quality (the discriminative power of the codebook), many measures have been proposed in the literature. Those include combinatorial techniques [153], and external cluster evaluation measures like F-measure [154], misclassification index (MI) [155], among others. Among the external evaluation measures, those based on information theory, like purity and conditional entropy, are independent of the size of the data set, the number of clusters and the clustering algorithms used. This provides information theory based measures a unique advantage over other

classes of measures [156, 157]. Following this reasoning, [61] proposed the use of purity and conditional entropy as evaluation measures for visual codebooks. However, both these measures improve with an increase in the number of clusters, up to a degenerate maximum where there are as many clusters as data points. Therefore, clustering evaluations based on these metrics are biased and score high on suboptimal solutions [156]. The rest of this subsection presents the details about entropy-based measures, their drawbacks, and the proposed metric.

Let $P = \{p_1, p_2, ..., p_C\}$ represent the probability distribution of the training descriptors belonging to C shape categories. Then the information conveyed by this distribution, entropy of P, is given by,

$$\text{Info(P)} = -\sum_{j=1}^{C} p_j \log_2 p_j \ , \tag{3.2}$$

$$p_j = N_j/N \tag{3.3}$$

where N_j is the number of data points belonging to class j and $N = N_1 + N_2 + ... + N_C$, is the total number of data points. After partitioning the data into K clusters, the entropy of each cluster E_i is given by,

$$E_i = -\sum_{j=1}^{C} p_{ij} \log_2 p_{ij}, \quad i = 1, 2, \cdots, K \tag{3.4}$$

where p_{ij} is the ratio of number of samples of class j in cluster i (n_{ij}) to the total number of samples in cluster i (n_i),

$$p_{ij} = n_{ij}/n_i \ . \tag{3.5}$$

The entropy of the codebook is the weighted average of the entropies of the K clusters,

$$\text{Info(P, K)} = \sum_{i=1}^{K} p_{c_i} E_i \tag{3.6}$$

where p_{c_i} is the ratio of the number of samples in cluster i (n_i) to the total number of samples (N),

$$p_{c_i} = n_i / N \ . \tag{3.7}$$

Thus the information gain, denoted as Gain(P, K), is defined as,

$$\text{Gain(P, K)} \ = \text{Info(P)} - \text{Info(P, K)} \ . \tag{3.8}$$

In order to maximize information gain, the entropy of the codebook Info(P, K) is to be minimized. This quantity goes to zero in an undesirable fashion when every sample or data point is treated as a cluster. In the machine learning domain, the induction of ID3 decision trees suffered from a similar problem and was rectified by normalizing the information gain using the split information [65]. The split information takes into account the number of data points in the clusters, and thus prevents over-fitting. By normalizing information gain using the split information of the codebook, defined in equation (3.10), the resultant term,

$$\text{GainRatio(P, K)} = \frac{\text{Info(P)} - \text{Info(P, K)}}{\text{SplitInfo(P, K)}} \tag{3.9}$$

can be maximized. In the context of clustering, split information is given by,

$$\text{SplitInfo(P, K)} = -\sum_{i=1}^{N} p_{c_i} \log_2 p_{c_i} \ . \tag{3.10}$$

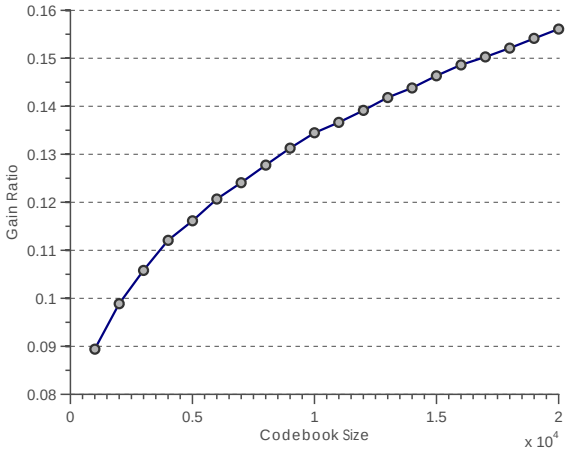


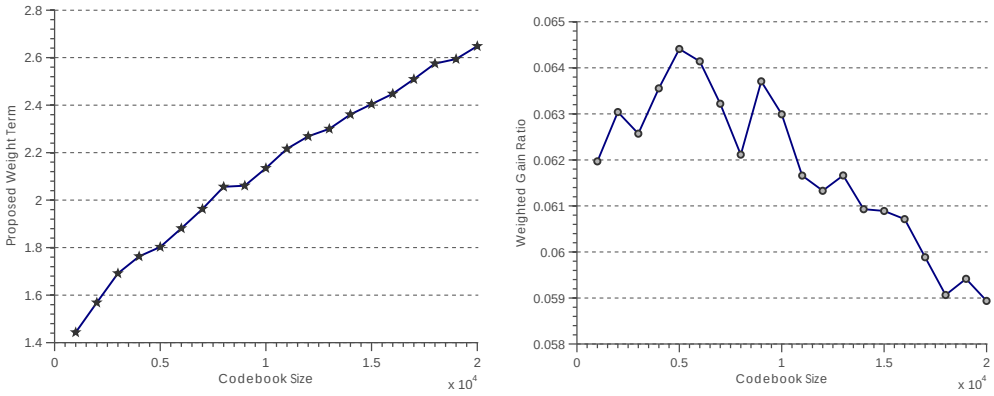
Figure 3.4: Gain ratio for codebook sizes up to 20,000.

Ideally, split information should increase significantly with increase in codebook size, and in turn, lead to a decrease in gain ratio before reaching a very high codebook size. However, gain ratio keeps increasing without attaining a maxima, up to a very high codebook size of 20000 (Fig. 3.4). This observation is supported by reference [158], which demonstrated that gain ratio is still biased in favor of attributes with large number of values. To address this problem, we propose the following metric which considers the 'physical size' of the clusters in the codebook.

$$\text{WeightedGainRatio(P, K)} = \frac{\text{Info(P)} - \text{Info(P, K)}}{\text{SplitInfo(P, K)} \times \text{VarianceRatio(P, K)}} \tag{3.11}$$

where the proposed 'weight term' is defined as,

$$\text{VarianceRatio(P, K)} = \frac{V_K}{V_{avg}} \ . \tag{3.12}$$



(a) Codebook Size vs. proposed weight term (b) Codebook Size vs. weighted gain ratio

Figure 3.5: Sample trends of the proposed weight term and metric for codebooks corresponding to Figure 3.4.

The variance of the codebook V_K is defined as,

$$V_K = \frac{{d_{c_1}}^2 + {d_{c_2}}^2 + ... + {d_{c_K}}^2}{K} \tag{3.13}$$

where d_{c_i} represents the Euclidean distance between the centroid of i^{th} cluster to the mean of all centroids. When the number of clusters is small, we can expect equation (3.13) to have a small value and increase as the number of clusters increases. The variance of a single cluster in the codebook is similarly defined as,

$$V_i = \frac{{d_{1i}}^2 + {d_{2i}}^2 + ... + {d_{n_i i}}^2}{n_i} \tag{3.14}$$

where d_{ki} is the Euclidean distance between the k^{th} member of cluster i to the cluster centroid. The average variance of the clusters is obtained by taking the mean value of all cluster variances:

$$V_{avg} = \frac{1}{K} \sum_{i=1}^{K} V_i \ . \tag{3.15}$$

As the codebook size increases, the average cluster size is expected to decrease. Therefore, the weight term (equation 3.12) increases in magnitude as the codebook size increases, as shown in Figure 3.5(a). Notice that 'weighted gain ratio' leverages on both the number of data points in a cluster (split information) and the size of each cluster (variance term) to account for the change in codebook size. Since it is possible to have many points in a cluster and still have a small cluster size, and vice versa, it is important to use both split information and the variance term together. In other words, the proposed weight term is expected to increase the rate of change of split information, and thus avoid very high codebook sizes. Thus, if we have a set of codebooks, it is possible to select one that maximizes 'weighted gain ratio', as illustrated in Figure 3.5(b).

3.2.3 Joint Learning Framework

If the extracted local features of the image are not discriminatory, then optimizing the codebook size is of no purpose. So, we formulate an iterative approach to feature learning and codebook size selection (refer to Algorithm 1). Using the initial LPT parameter settings, the first iterative step of the joint learning framework selects the codebook with maximum weighted gain ratio (equation (3.11)). For the obtained codebook size, the second step selects a codebook which maximizes gain ratio (equation (3.9)) among codebooks with different values of LPT r_{max}. These two steps are iterated until there is convergence of codebook size and LPT maximum radius. Note that 'weighted gain ratio' is not used for the second iterative step, because of fixing the codebook size from the output of the first step. The output of the joint learning framework is a codebook of a particular size and LPT r_{max}, which is then used to represent the training/testing images.

Algorithm 1 Joint learning algorithm

1: Set the feature parameters $n_r = 14, n_w = 30, r_{min} = 2$ and initialize r_{max} using equation (3.1).
2: **repeat**
3: Choose a codebook with size $K \in \{1000, 2000, ...20000\}$ using weighted gain ratio.
4: Fix codebook size K from previous step.
5: Choose a codebook with LPT $r_{max} \in \{5, 10, \cdots, 125\}$ using gain ratio.
6: Fix LPT r_{max} from previous step.
7: **until** no change in K, r_{max}
8: **Output:** Codebook of size K using LPT features with r_{max}.

In the next subsection, details about the proposed histogram representation are presented.

3.2.4 Contextual Information

The bag-of-words histogram representation discards the spatial relationship between the local features. As mentioned earlier in Chapter 1, spatial pyramid matching [50] was proposed to encode coarse, mid-level spatial relationships between the local features. However, due to its high-dimensional histogram representation, some previous works have opted for compact representations [51], or Markov stationary features [57], or higher order spatial co-occurrence statistics [96]. Higher-order statistics can yield richer information, but in applications involving sparse sampling of the image (approx. 3% of the total image pixels in this work), it may not be readily derivable. On the other hand, the Markov stationary features (MSF) proposed in [57] provides an attractive alternative for encoding spatial information using the spatial co-occurrence matrix [58]. Nevertheless, the stationary distribution of MSF is a computationally intensive and an indirect way to capture information from the spatial co-occurrence matrix. Notice that the problem of image processing is similar to language processing,

because the image can be considered as an article written using many "visual" words of the codebook. Inspired by this idea, we interpret each entry in the spatial co-occurrence matrix as the pairwise occurrence count of the codewords, or in other words, bi-gram count. The following paragraphs present the details about the proposed bi-gram extraction procedure.

For each training/testing image, feature extraction followed by vector quantization enables each local descriptor to be assigned to one of the K visual words. So, let us define an image I_{ind}, having the word indices - $\{1, 2, \cdots, K\}$ - as pixel values. The word indices simply represent the K visual words, $S = \{c_1, c_2, ...c_K\}$, assigned to the LPT descriptors during vector quantization (section 3.2.5). In this work, the boundaries of the binary shape are assigned to a particular index of the visual word and other locations, where local features are not extracted, are set to zero. Therefore, each pixel $I_{ind}(x, y)$ of an $m \times n$ index image takes one of the values in the set $\{0, 1, 2, \cdots, K\}$. The spatial co-occurrence matrix is created by calculating how often a pixel value i ($i \neq 0$) occurs adjacent to a pixel with the value j ($j \neq 0$). We denote the co-occurrence matrix as $\mathbf{C} \in \mathrm{R}^{\mathrm{KxK}}$, in which each entry is computed as follows.

$$C(i,j) = \sum_{x=1}^{n} \sum_{y=1}^{m} \#(I_{ind}(x,y) = i, I_{ind}(x_n, y_n) = j) \tag{3.16}$$

where every pixel location (x, y) and its immediate neighbors (x_n, y_n) satisfying $0 < \sqrt{(x_n - x)^2 + (y_n - y)^2} \leq \sqrt{2}$, are inspected to count the number of i-j pairs. Simply put, each entry $C(i, j)$ in the spatial co-occurrence matrix records the number of times a pair of neighboring local descriptors get assigned to c_i and c_j, which are any two of the K visual words.

The corresponding transition matrix $\mathbf{T} \in \mathrm{R}^{\mathrm{KxK}}$ is defined as,

$$T(i,j) = \frac{C(i,j)}{\sum_{k=1}^{K} C(i,k)}. \tag{3.17}$$

Since the spatial co-occurrence matrix is row normalized to obtain the transition matrix (equation 3.17), each element of the transition matrix represents the pairwise occurrence probability of the codewords, or in other words, bi-gram occurrence probability. So to extract discriminatory features, we select bi-grams with high probability from all the transition matrices of the training data, and subsequently discard those that occur across different shape categories. As a result, it is possible to retain a unique signature for each category in the histogram representation, and at the same time, reduce computational load. The selected bi-grams are termed as 'class-unique bi-grams', because they appear with high probability within training images of a single shape category. After selecting the class-unique bi-grams, the spatial co-occurrence matrix characterizes their frequency for each training/testing image. This procedure is described below using an example.

The algorithm for extracting class-unique bi-grams is illustrated in Fig. 3.6, in which three sample codewords denoted as A, B, and C are used to represent the training images from two classes as 3 by 3 transition matrices (refer to [57] for a visual description of obtaining the transition matrix from the image). Notice that the spatial co-occurrence matrix can be easily derived from the transition matrices in Fig. 3.6, by simply considering the numerator term in each entry. For instance, the AA bi-gram in the first matrix of the monkey class has occurred twice, the AB bi-gram thrice, and so on. Thus, in total, the codeword A has

been assigned five times to a local descriptor, whose neighboring descriptors have been assigned to either A itself or the codeword B. By setting a threshold of 0.4 for each transition matrix, the circled entries represent those bi-grams which are above this probability threshold. The potentially confusing bi-grams (BA and BC), which appear in both the classes with a high probability, are discarded and the remaining seven entries - AA, AB, AC, BB, CA, CB, CC - are further investigated. Since the spatial co-occurrence matrix is symmetrical, duplicate entries like AC and CA are singled out and either of them are kept. In addition, entries AB and CB are removed because their symmetrical counterparts BA and BC were discarded. Finally, a 4-dimensional histogram of bi-grams using AA, AC, BB and CC is created using the corresponding spatial co-occurrence matrix. Note that it is easy to implement this algorithm using the MATLAB commands - *fliplr*[1] (check symmetry of the matrix indices) and *unique*[2] (extract class unique bi-grams and remove duplicate entries).

3.2.5 Vector Quantization and Classification

A training/testing image is quantized into K histogram bins, i.e., the local features extracted from an image are individually matched to the nearest visual word using Euclidean distance and the frequency of each word creates the K-dimensional histogram representation. Let the number of class-unique bi-grams be N_{bi}. The normalized bag-of-words representation is concatenated with the normalized histogram of bi-grams, to produce a vector of dimension - $(K + N_{bi})$. Besides the bi-gram features, a 2×2 image grid is used to capture mid-level spatial information. Each of the four histograms from the 2×2 grid is normalized

[1] http://www.mathworks.com/help/matlab/ref/fliplr.html
[2] http://www.mathworks.com/help/matlab/ref/unique.html

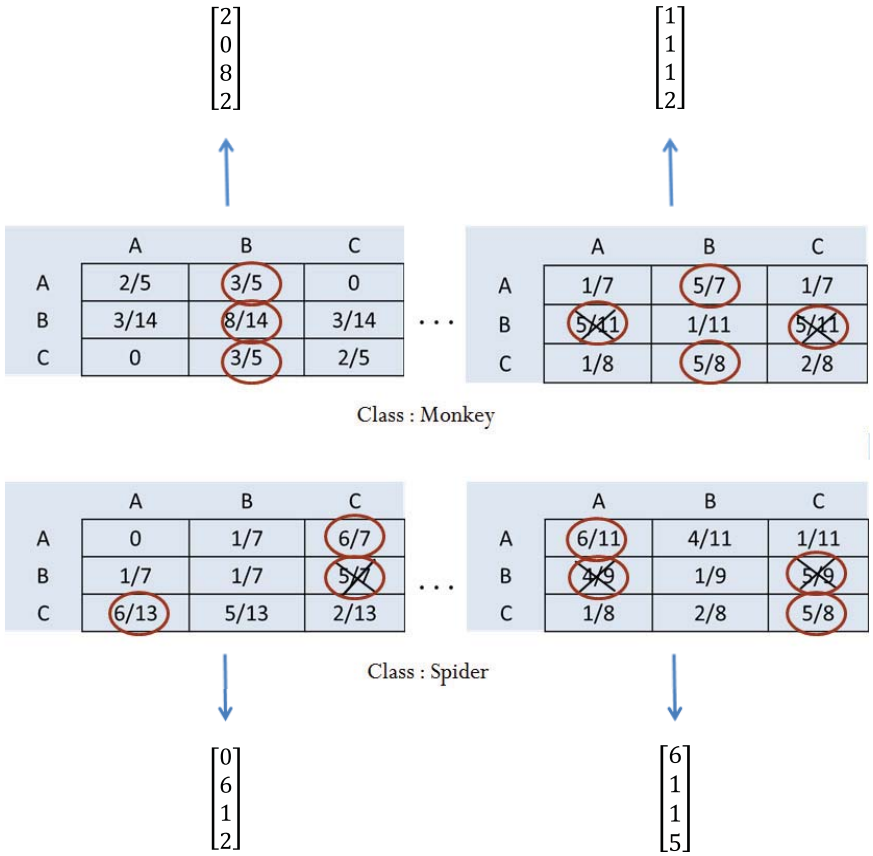


Figure 3.6: An example of the bi-grams extraction procedure from transition matrices of the training data.

separately and concatenated together. In turn, the 4K-dimensional vector using the 2×2 grid is concatenated with the $(K + N_{bi})$-dimensional vector to form the final $(5K + N_{bi})$-dimensional representation of each image. The classifier used is the SVM implementation of VLFeat in their bag-of-words application [152].

3.3 Experiments and Results

We tested our shape classification system on the animal shapes database introduced by Xiang et al. [151], which consists of 2000 binary shapes of 20 animal categories with 100 shapes for each category. The dataset poses several challenges, such as large intra-class variations, strong inter-class similarities among

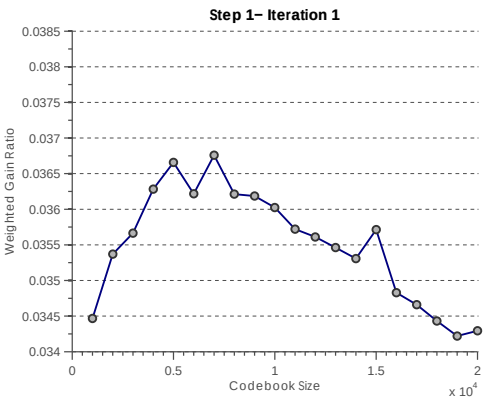
some categories, viewpoint changes and occlusion (Fig. 3.7). Following the protocol in [49,151,159,160], the database was randomly split into half for training and half for testing. Our experiments were run on HP Xeon Two Sockets Quad-Core 64-bit Linux clusters with 200 GB memory limit.

3.3.1 Joint Learning Results

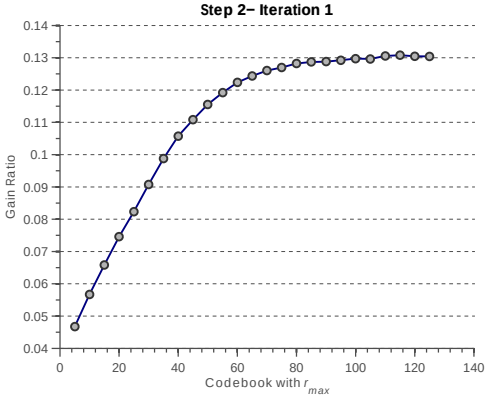
The steps taken by the joint learning framework are shown in Fig. 3.8. After LPT r_{max} was initialized using equation (3.1), weighted gain ratio peaked at a codebook size of 7000 in the first iteration (Fig. 3.8(a)). During the next phase to select r_{max}, gain ratio showed an increasing trend as expected, and eventually saturated for codebooks with a very high LPT r_{max} (Figure 3.8(b)). Therefore, a relative change threshold of 1% was used for selecting a codebook with a moderate LPT $r_{max} = 75$, which corresponds to the codebook with the maximum gain ratio before reaching the threshold. After setting $r_{max} = 75$, weighted gain ratio peaked at a lower codebook size of 5000 in the next iteration (Fig. 3.8(c)). This is possibly due to the selection of a better LPT parameter from the first iteration. In the next phase, the same maximum radius was selected based on a relative change threshold of 1% for (Fig. 3.8(d)). Therefore, a couple of iterations were sufficient to converge on the final codebook size to be 5000 with LPT $r_{max} = 75$. Next, we investigate whether the parameters selected by the joint learning algorithm give high classification accuracy compared to a wide range of settings.

Figure 3.9 shows the classification accuracy obtained with different codebook sizes and LPT r_{max}. For the bag-of-words model, codebooks with LPT $r_{max} > 40$ give higher classification accuracy compared to the codebook with

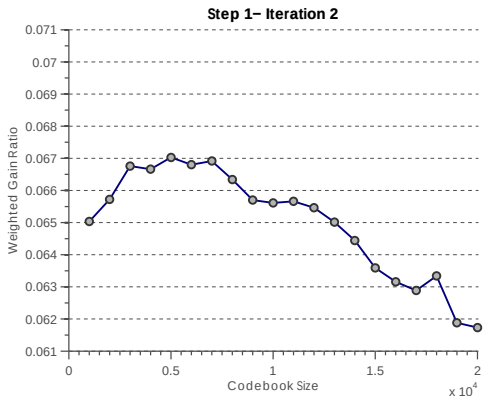
Figure 3.7: Samples from the animal shapes dataset.



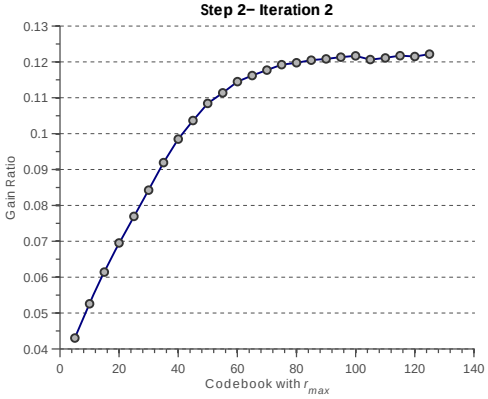
(a) LPT r_{max} initialized using equation (3.1)



(b) Codebook size set to 7000

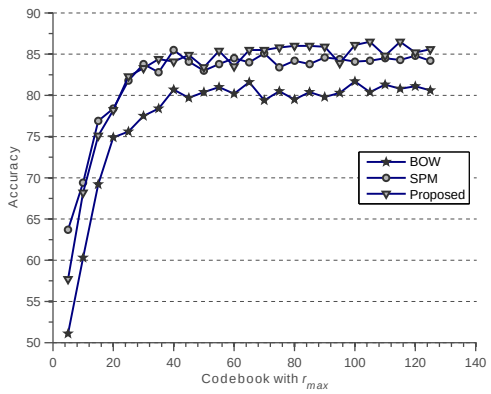


(c) LPT r_{max} set to 75

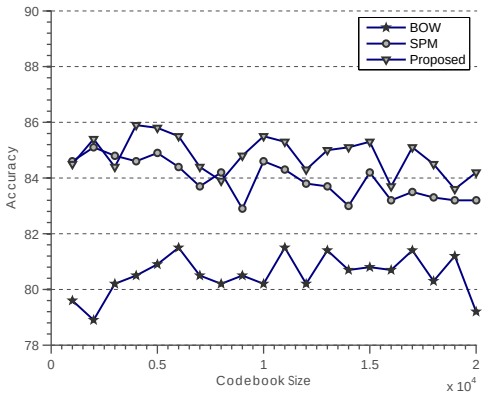


(d) Codebook size set to 5000

Figure 3.8: Results from the joint learning framework over two iterations.



(a) Fixed codebook size = 5000



(b) Fixed LPT $r_{max} = 75$

Figure 3.9: Classification accuracy obtained with three methods for different LPT maximum radius and codebook sizes (%); Legend: BOW - Bag of Words for LPT, SPM - Spatial Pyramid Matching for LPT, Proposed - LPT (1×1 & 2×2) + Contextual Information

$r_{max} = 30$ initialized using equation (3.1) (refer to Fig. 3.9(a)). Although the codebook with $r_{max} = 75$ did not give the highest accuracy, the effectiveness of the joint learning strategy is clearly evident. A similar trend can be observed for the spatial pyramid approach using codebooks with different maximum radius of the log-polar transform. For the proposed method, the selected codebook with LPT $r_{max} = 75$ provides a 2% boost in classification accuracy compared to the codebook initialized with LPT $r_{max} = 30$ (Fig. 3.9(a)). By inspecting Fig. 3.9(b), it is clear that the codebook size selected using the joint learning algorithm (5000) gives a high classification accuracy, which is close to the highest obtained for the bag-of-words model. Larger codebooks may provide higher classification accuracy for the bag-of-words representation, but it can drop considerably as seen from the trend towards a codebook size of 20,000. In contrast, SPM provides good classification accuracy for codebook sizes only up to 6000 and suffers from the high-dimensionality of the histogram (21K) for higher codebook sizes (refer to Fig. 3.9(b)). In comparison to SPM, the incorporation of bi-gram features in the bag-of-words model provides higher classification accuracy and a relatively stable trend for codebook sizes up to 10,000 (Fig. 3.9(b)). In summary, we have shown that there is a positive correlation between the model parameters selected using the joint learning algorithm and the classification accuracy.

3.3.2 Classification Results

To demonstrate the effectiveness of the proposed histogram representation, the performance of our method is compared with four baseline methods in Table 3.2, where "Bi-gram" is the representation with the global BoW histogram coupled with the class-unique bi-grams, and "Proposed" is the 2×2 grid rep-

Table 3.2: Performance comparison of the proposed methods with four baseline methods (%).

Alg.	BoW	MSF [57]	Triv.MSF [161]	SPM [50]	Bi-gram	Proposed
Acc.	81.1	81.1	81.5	84.8	83.5	86.0

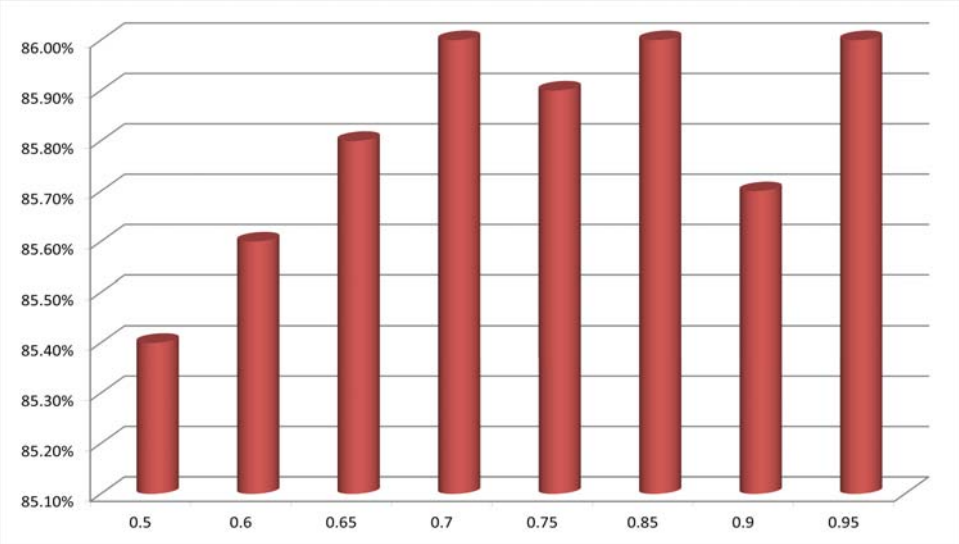


Figure 3.10: Effect of changing the probability threshold for the transition matrix on classification accuracy of the proposed method.

resentation added to the "Bi-gram" model. Note that the Markov stationary feature (MSF) [57] performs on par with the bag-of-words model. In contrast, we show that the spatial co-occurrence matrix can be exploited in a much simpler way to boost the classification accuracy ("bi-gram"). The results reported for the proposed method uses a probability threshold of 0.7 for the transition matrix, which resulted in 11,275 class unique bi-grams. Fig. 3.10 shows the effect of varying the probability threshold on the classification accuracy. It can be noted that the classification accuracy does not change significantly for different threshold values. This phenomenon can be attributed to the selection of frequently occurring bi-grams that appear within a single shape category of the training set.

Table 3.3 compares the proposed method with previous works using the an-

imal shapes dataset. It is clear that the proposed method significantly outperforms the state-of-the-art algorithms while still using a low-dimensional histogram representation. In comparison to SPM, which uses a 21K-dimensional histogram representation, the proposed method uses only a 7K-dimensional histogram representation. Note that the training and testing set were generated using a random database split – half for training and half for testing – as in [49, 151, 159, 160]. So, one may argue that the high classification result is possibly due to the "random" nature of training and testing. In order to further demonstrate the virtue of the proposed histogram representation and the local descriptor, ten-fold cross validation was done with a codebook size of 5000 and LPT $r_{max} = 75$, which resulted in classification accuracy of 87.8%. Thus, conclude that the reported classification accuracy (86.0%) is due to a genuine improvement of the bag-of-words model.

Besides cross validation, we can manually verify whether the system makes reasonable mistakes by inspecting the confusion matrix (Fig. 3.11). We observe that animals with distinct visual attributes like spider, butterfly, hen, elephant, duck, deer, and horse contribute to a total error of just 2%. The potentially confusable ones, which share similar physical attributes, like dog, leopard, cat, and mouse pose a greater problem to the classification system. The lack of depth information for the shapes creates ambiguity between different shape categories, even for humans, especially among four-legged animals and aquatic species (see Fig. 3.7). Therefore, we conclude that the proposed shape classification framework is capable of high performance under challenging conditions like scale, rotation and strong viewpoint variations.

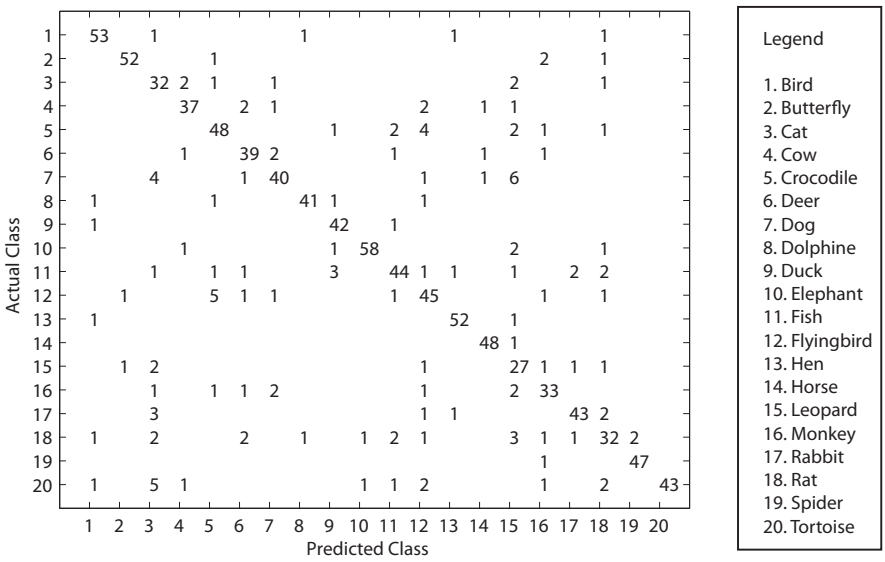


Figure 3.11: Confusion matrix for the best result of our system on the animal shapes dataset.

Table 3.3: Performance comparison of the proposed method with previous works (%).

Method	Accuracy
IDSC [162]	73.6
CS [140]	71.7
CS&SP [151]	78.4
CS&SP&IDSC-F [151]	78.7
CS&SP-DP [159]	80.7
Shape Tree [160]	80.0
HOG-SIFT BoW [49]	80.4
Proposed Method	**86.0**

Comparing Table 3.2 and Table 3.3, LPT based bag-of-words ("BoW" in Table 3.2) slightly outperforms the bag-of-words representation using the popular feature descriptor SIFT [49]. However, the comparison may not be fair because of differences in implementation and other parameters. Hence, in the next subsection, we describe our implementation of a SIFT based bag-of-words model and compare it with the proposed framework.

3.3.3 Comparison with SIFT

Ignoring scale selection, many works have found that SIFT sampling at multiple scales performs well in object classification systems using a bag-of-words framework, as noted in [55]. For establishing a fair comparison with the proposed method, we replace log-polar transform feature extraction step with SIFT descriptor at multiple scales, while all other components remain unchanged. The chosen scales are four, six, eight and ten, following the publicly available implementation of VLFeat's object classification system [152]. The codebook size was chosen to be 3000 using 'weighted gain ratio' from a range of codebook sizes – 1000, 2000,..., 20000. Using the above settings, we obtained an accuracy of 80.2% for the SIFT-based bag-of-words implementation, which is markedly similar to the accuracy of 80.4% obtained in [49]. Therefore, we conclude that the accuracy obtained using the proposed LPT-based method is significantly higher than SIFT-based bag-of-words for binary shape classification. In the following subsection, we compare the proposed LPT local feature with other well-established shape descriptors.

3.3.4 Comparison with Fourier Descriptors

Fourier descriptors have long had a good reputation for shape representation and retrieval. Thus, we consider two popular global Fourier descriptors - centroid distance signature (also known as 1-D Fourier descriptor) [163] and generic Fourier descriptor (GFD) [164] - for comparison with the global LPT approach. Note that the centroid distance signature can be readily implemented using DIPUM toolbox [165] and GFD's polar transform using the command -

Table 3.4: Comparison of the global LPT shape descriptor with Fourier shape descriptors in terms of the classification accuracy on the animal shapes dataset (%).

	CentroidDistance	GFD	GlobalLPT
Accuracy	35.10	52.80	**53.70**

cart2pol - in MATLAB. Table 3.4 shows the comparison between these methods and the global LPT approach, described earlier in section 3.2.1. Notice that we have not compared the Fourier descriptors with the proposed local LPT framework, because they were proposed as global shape descriptors much before the bag-of-words model became the dominant classification framework. Therefore, we extend GFD as a local descriptor and compare it with the proposed LPT local shape descriptor. It should be noted that generic Fourier descriptor is a closely related work, because it uses LPT's counterpart - polar transform - for obtaining the shape descriptor. In other words, using GFD as a local descriptor is equivalent to replacing log-polar transform with polar transform in the proposed framework, which makes for a very interesting comparison. The selection of the parameters for the GFD local descriptor is explained below.

The size of the polar grid was chosen to be the same as LPT's grid size, i.e., 14 rings and 30 wedges. The minimum and maximum radius were chosen to be 2 and 40, respectively. Note that the maximum radius of the polar grid was chosen exhaustively to give the best classification accuracy while the codebook size was chosen to be 3000 using 'weighted gain ratio'. Other codebook sizes were also investigated to see if better classification accuracy could be achieved. Finally, the best settings for the GFD local approach scored an accuracy of 83.7%, whereas the proposed LPT approach achieved 86% classification accuracy

on the animal shapes dataset. Thus, we conclude that log-polar transform, as a local descriptor for representing binary shapes, has a clear edge over GFD. Since GFD uses equidistant polar sampling, only rotations in the Cartesian domain are converted to translations in the angular axis (scaling becomes multiplicative). Whereas in the log-polar case, both scale and rotation changes in the Cartesian domain are transformed into translations along the new axes. This invariance to scale and rotation changes gives a clear edge to LPT over GFD, as demonstrated in the experiments above.

3.4 Summary

In this chapter, we proposed a robust shape classification system, which can handle scale, rotation and strong viewpoint variations, using log-polar transform as a local feature in the bag-of-words framework. In the proposed framework, contextual information was incorporated using a novel method to extract bi-grams from the spatial co-occurrence matrix. We showed that the histogram of bi-gram representation greatly improves on the standard bag-of-words model and its offshoot, spatial pyramid matching. Besides the above contributions, a novel metric was proposed to select an appropriate codebook size in the bag-of-words model. The selected codebook size was shown to give a high accuracy, compared to a wide range of codebook sizes. Furthermore, the proposed metric for codebook selection is generic, and thus can be used for any clustering quality evaluation. Lastly, we proposed a joint learning framework for learning features in a data-driven manner from the training set. The procedure iterates between setting the codebook size and the maximum radius of the log-polar transform,

which was demonstrated to be effective in improving the classification accuracy without the requirement of manual parameter tuning. We tested our algorithm on a challenging shape database and achieved a 6% increase in accuracy compared to state-of-the-art algorithms in the literature.

Our next work would be to extend this framework for object classification in grayscale images. Direct application of log-polar transform as a local feature on every pixel of the image, as in dense SIFT, may not be ideal in terms of computational load or accuracy. An efficient way to perform keypoint detection and feature extraction is required, at the least, to be on par with well-established local descriptors like SIFT, LBP, etc.

Chapter 4

Cue-based Unseen Object Categorization using Optimized Visual Dictionaries

4.1 Introduction

After designing the shape classifier in the previous chapter, we now consider the more general case of classifying objects from grayscale images. The main idea is to use both appearance and shape cues to complement the information available from different object cues, such as structure, texture and shape. The integration of these object cues is done using the bag-of-words model by constructing an optimized visual vocabulary for each cue. Then, the histogram representation of the log-polar encoded local features from each cue are combined to obtain the final image representation. We evaluate the proposed object classification system by adopting the leave-one-object-out protocol on a dataset

with the segmentation ground truth for each object, which provides an ideal way to quantify the quality of the extracted shape.

The rest of this chapter is organized as follows. We introduce the details of our proposed methods in section 4.2, which includes the feature extraction module, the codebook optimization algorithm, and the multi-cue representation. Next, we evaluate the proposed method on the ETH-80 dataset and present the experimental results and discussion in section 4.3. Finally, we conclude this chapter in section 4.4.

4.2 Cue-based Bag-of-Words Model

We adopt the bag-of-words framework consisting of four main stages: keypoint detection, feature extraction, vector quantization, and classification. For classifying the grayscale images, we extract three cue images representing the structure, texture and shape of the object. For the grayscale appearance cues (structure and texture), keypoint detection is done using the Rudin-Osher-Fatemi image denoising method [82]; for the extracted binary image, the keypoints are simply the boundary points of the shape. Feature extraction involves sampling the cue images at the keypoints, using log-polar transform. The set of descriptors from each cue of the training images are collectively used to obtain a codebook. In this case, three codebooks will be generated using the training set. Then, the quantization step is the histogram representation of each cue image, using the respective codebooks generated in the previous step. Subsequently, the histograms of the training images are formed by a *late fusion* step, i.e., the histograms obtained for all the cues are concatenated to form the final representation of each

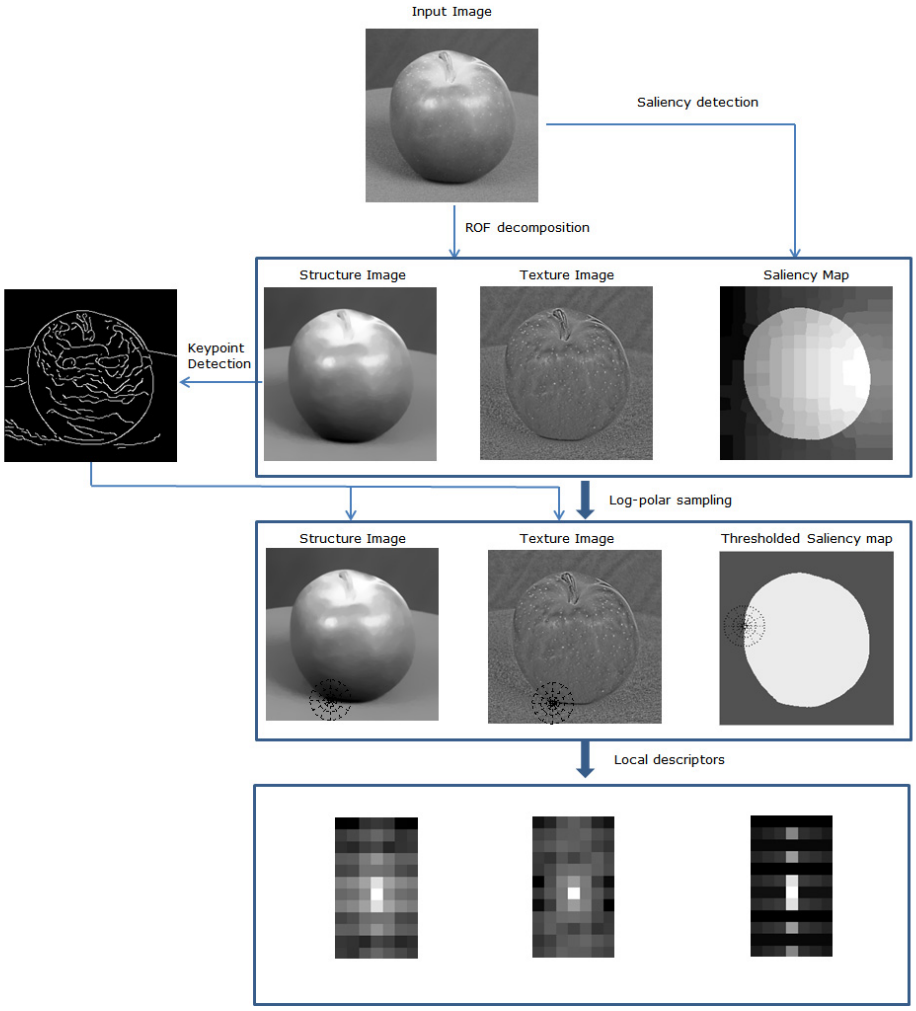


Figure 4.1: Feature extraction for cue-based object categorization. On one hand, the input image is decomposed into structure and texture using the Rudin-Osher-Fatemi method; on the other hand, saliency detection is performed on the input image to obtain a saliency map, which is further binarized using the Otsu method. Using log-polar transform, the keypoints obtained from the structure image are used to sample the grayscale appearance cues (structure and texture) while the binary shape is sampled at its boundaries.

image. Finally, the histograms of the training images are used to train an SVM classifier. During testing, the codebook construction step is bypassed, and a test image is simply represented using the codebooks and classified using SVM. Figure 4.1 illustrates the details of the proposed cue-based feature extraction step.

4.2.1 Keypoint Detection

We define the keypoints as locations on the image with a distinctive appearance with respect to its neighboring pixels. Therefore, to deal with noise, we first use the Rudin-Osher-Fatemi model to obtain the denoised image, and then use the Canny edge detector to obtain the keypoints. The ROF denoising model is based on the principle that images with excessive and possibly spurious details have high total variation. In other words, the integral of the absolute gradient of the signal will be high. Accordingly, by reducing the total variation of the image, subject to being a close match to the original image, unwanted details can be removed whilst preserving important ones, such as edges. For the input grayscale image $v(\mathbf{x}) : \Omega \subset \mathrm{R}^2 \rightarrow \mathrm{R}$, the denoised image $u(\mathbf{x})$ is given as the solution of

$$\min_u \int_\Omega \left\{ \frac{1}{2\theta}(u-v)^2 + |\nabla u| \right\} d\mathbf{x}. \tag{4.1}$$

where θ is a small constant, such that u is a close approximation of v. To solve equation 4.1, an efficient iterative scheme that is globally convergent was proposed in [166]. The solution is based on gradient descent and subsequent re-projection using the dual-ROF model. Since this algorithm is a core component of our framework, we reproduce the relevant results below.

Proposition 4.1. The solution of equation 4.1 is given by

$$u = v + \theta \ \mathbf{div} \ \mathbf{p}, \tag{4.2}$$

where the dual variable $\mathbf{p} = [p_1, p_2]$ is iteratively defined as,

$$\tilde{\mathbf{p}}^{n+1} = \mathbf{p}^n + \frac{\tau}{\theta}(\nabla(v + \theta \ \mathbf{div} \ \mathbf{p}^n)) \text{ and} \tag{4.3}$$

$$\mathbf{p}^{n+1} = \frac{\tilde{\mathbf{p}}^{n+1}}{\max\{1, |\tilde{\mathbf{p}}^{n+1}|\}} \ , \tag{4.4}$$

where $\mathbf{p}^0 = \mathbf{0}$ and the time step $\tau \leq \frac{1}{4}$.

The denoising method described above has advantages over simple techniques such as Gaussian smoothing or median filtering, due to the fact that simple filtering techniques reduce noise, but at the same time smooth away edges to a greater or lesser extent. In contrast, total variation denoising is remarkably effective, even at low signal-to-noise ratios, at preserving edges while removing noise in flat regions (see the structure image in Fig. 4.1). To extract the textural part, the difference between the original image and the denoised image is computed as, $v(\mathbf{x}) - \alpha u(\mathbf{x})$, where the blending factor α is set to be 0.95 as in [81]. The Canny edges extracted from the structural part are used as keypoints for the grayscale texture image as well.

4.2.2 Feature Extraction

For each boundary point in the extracted binary shape, log-polar sampling is accompanied by computing its Fourier transform modulus. Subsequently, the local descriptor is obtained by converting the two-dimensional Fourier transform

output into a vector and performing normalization using the Euclidean norm. For sampling the grayscale cues using log-polar transform, directly obtaining its Fourier transform modulus is not suitable, because noise, small appearance changes and non-uniform lighting conditions can severely affect the invariant properties. An alternative is to use bandpass filters at multiple scales and extract only the high energy Fourier transform components, as done by [36]. In contrast, we use the ROF denoising method and encode the resulting grayscale cue values using log-polar transform. Next, we present the LPT parameter settings for encoding the grayscale cues and the binary shape cue.

For the grayscale appearance cues, we set $r_{min} = 1$ and $r_{max} = 7$, which would roughly encode a 14 by 14 patch around each keypoint. For the binary shape, we set $r_{min} = 1$ and $r_{max} = 40$, following the recommendations in [84]. Furthermore, the number of rings (n_r) is set to 7 and number of wedges (n_w) is set to be 12 for sampling both the binary shape and the grayscale cues.

4.2.3 Codebook Optimization

Similar to the earlier chapter, the descriptors obtained from the training images are collectively used to obtain a codebook, but using VLFeat's [152] Approximate Nearest Neighbor (ANN) K-means algorithm for faster optimization. The codebook is simply the collection of the cluster centroids obtained using K-means.

Let $P = \{p_1, p_2, ..., p_C\}$ represent the probability distribution of the training descriptors belonging to C shape categories. Then the information conveyed by

this distribution, entropy of P, is given by,

$$\text{Info(P)} = -\sum_{j=1}^{C} p_j \log_2 p_j \ , \tag{4.5}$$

$$p_j = N_j/N \tag{4.6}$$

where N_j is the number of data points belonging to class j and $N = N_1 + N_2 + ... + N_C$, is the total number of data points. After partitioning the data into K clusters, the entropy of each cluster E_i is given by,

$$E_i = -\sum_{j=1}^{C} p_{ij} \log_2 p_{ij}, \quad i = 1, 2, \cdots, K \tag{4.7}$$

where p_{ij} is the ratio of number of samples of class j in cluster i (n_{ij}) to the total number of samples in cluster i (n_i),

$$p_{ij} = n_{ij}/n_i \ . \tag{4.8}$$

Those clusters with a low entropy will have members from a few object categories, and would play a crucial role in obtaining a discriminative image representation after vector quantization. On the other hand, clusters with high entropy have members from many object categories, and this makes it a suspicious candidate for providing a useful image representation. However, some clusters, with moderately high entropy, would provide to be useful if the categories share similar features. To account for this case, we use cross-validation to select the clusters within a range of entropy values while removing potentially disadvantageous and

redundant clusters. The rescaling of entropy values is given as,

$$E_i^{new} = (b - a) \times \frac{(E_i - m)}{(M - m)} \tag{4.9}$$

where the values of $a = 0$ and $b = 1$ are used to rescale the entropy values between $[0, 1]$, and m and M represent the minimum and maximum entropy values out of all the clusters in the codebook, respectively. Thus, a threshold is varied between 0 and 1 to obtain the set of clusters that give the best performance during cross-validation.

4.2.4 Vector Quantization and Classification

For each object cue, a training/testing image is quantized into K_i histogram bins ($i = 1, 2,$ and 3 for structure, texture and shape respectively), i.e., the local features extracted from a cue image are individually matched to the nearest visual word of the respective codebook using Euclidean distance and the frequency of each word creates the K_i-dimensional histogram representation. Besides the global bag-of-words features, a 2×2 image grid is used to capture mid-level spatial information. Each of the four histograms from the 2×2 grid is normalized separately and concatenated together. In turn, the $4K_i$-dimensional vector using the 2×2 grid is concatenated with the K_i-dimensional vector to form the $5K_i$-dimensional histogram representation for each cue. Finally, the individual cue representations are concatenated to form the image representation of dimension $(5K_1 + 5K_2 + 5K_3)$. The classifier used is the SVM implementation of VLFeat in their bag-of-words application [152].

Figure 4.2: Sample images from the ETH-80 dataset

4.3 Experiments and Discussion

We tested our object classification system on the ETH-80 dataset, which was introduced by Leibe and Schiele [167] to specifically test for unseen object classification. The ETH-80 image dataset consists of 80 objects categorized into 8 classes, namely apple, pear, tomato, cow, dog, horse, cup and car (see Fig. 4.2). Each object is captured under 41 different viewpoints, and the testing protocol is to classify each object under all viewing angles while the rest of the objects are considered for training. Thus, classification is done for a total of 80 times for all the images in the dataset. Our experiments were carried out on HP Xeon Two Sockets Quad-Core 64-bit Linux clusters with 72 GB memory limit.

Table 4.1: Classification accuracy comparison of the proposed method with previous works (%).

Method	Accuracy
Color histogram [167]	64.86
PCA gray [167]	82.99
PCA masks [167]	83.41
SC&DP [167]	86.40
IDSC&DP [162]	88.11
IDSC&Morphology [168]	88.04
Height function [169]	88.72
Robust symbolic [144]	90.28
Kernel-edit [143]	91.33
BCF [76]	91.49
Proposed Method	**97.13**

4.3.1 Classification Results on the ETH-80 Dataset

We compare the performance of the proposed method to many earlier works on ETH-80 in Table 4.1. Our cue-based bag-of-words approach outperforms the previous state-of-the-art method in [76] by a large margin. In our experiments, we did not make use of the segmentation ground truth available in the dataset, whereas [76] is a shape classification framework which makes use of all ground truth shapes to report the result. On the other hand, some earlier works like [167] only use the color or gray level information to report the results on the ETH-80 dataset. Note that irrespective of the information used by the previous methods, all of them follow the leave-one-object-out protocol, and hence, comparison of the final accuracy is valid. Next, we show the individual performance of the structure, texture and shape cues, and also demonstrate the necessity of multiple cues for high classification performance.

Table 4.2 shows the performance of the individual object cues in comparison the performance of the proposed multiple cue representation. Clearly, using the original grayscale pixel values results in inferior performance when compared

Table 4.2: Classification accuracy of individual object cues in comparison with the accuracy of the proposed method (%).

Alg.	Grayscale	Structure	Texture	Str-Tex	Shape	Proposed
Acc.	84.97	86.80	88.87	92.53	92.25	97.13

to the usage of any individual object cue. Moreover, only when the structure and texture cues are combined together, the performance is as good as using the shape cue, which reaffirms the observations in the literature regarding the importance of shape cues for object recognition. Overall, the proposed method of combining all three cues leads to a very significant improvement in classifying unseen objects from a known category.

Table 4.3 and 4.4 shows the confusion matrix of the classification system that utilizes the shape and the structure-texture cue, respectively. It is clearly evident that without the grayscale appearance cues, distinguishing the round shapes of tomato and apple is difficult (see the first two shapes in the last row of Fig. 4.3), whereas classifying the animal shapes is difficult without the shape information. Naturally, a classification system that can leverage the benefits of both grayscale structure-texture and binary shape cues will perform much better than those using them separately, as shown in Table 4.5.

Figure 4.3 shows sample shapes extracted using the salient object detection algorithm. Although the extracted shapes are imperfect, we have achieved high classification performance since the proposed framework uses local features. The quality of the extracted shapes depends upon the saliency algorithm and the thresholding method. Since the saliency algorithm employed is one of the state-of-the-art solutions, we only focused on improving the shapes using different thresholding methods.

The quality of the extracted shape can be quantified using precision and recall. The two quantities are defined as, (a) *Precision*: The percentage area of the extracted shape that overlaps with the ground truth. (b) *Recall*: The percentage area of the ground truth that is contained within the shape. Mathematically,

$$P = \frac{|R_s \cap R_g|}{|R_s|}, \tag{4.10}$$

$$R = \frac{|R_s \cap R_g|}{|R_g|}, \tag{4.11}$$

where R_s is the extracted shape and R_g is the ground truth shape. Subsequently, average precision and average recall are calculated by obtaining the arithmetic average of the precisions and recalls obtained for all the shapes.

Table 4.6 lists the average precision, average recall and F-score (equation 4.12) of the shapes extracted using different thresholding algorithms. Out of all the thresholding algorithms developed from the early 60's to the 90's, the popular Otsu's method performs the best in terms of F-score. Other notable ones that have a high F-score are the entropy based method developed by Kittler and Illingworth [170], Prewitt and Mendelsohn [171]'s analysis of cell images, and the popular Tsai's moment preserving thresholding method [172]. In the next subsection, we present the results of the codebook optimization procedure.

$$F_{score} = \frac{2PR}{P + R} \tag{4.12}$$

Table 4.3: Confusion Matrix (%) for the shape cue on the ETH-80 dataset.

	Apple	Car	Cow	Cup	Dog	Horse	Pear	Tomato
Apple	84.39	0	0	0 .24	0	0	0	15.37
Car	0	97.80	0.49	0	0.49	1.22	0	0
Cow	0	4.39	86.34	0.49	3.17	5.61	0	0
Cup	0.24	0	0	99.52	0	0	0	0.24
Dog	0	0.49	3.41	0	93.17	2.93	0	0
Horse	0	0.49	3.17	0	3.66	92.68	0	0
Pear	0	0	0	0	0	0	100	0
Tomato	15.85	0	0	0	0	0	0	84.15

Table 4.4: Confusion Matrix (%) for the structure-texture cue on the ETH-80 database.

	Apple	Car	Cow	Cup	Dog	Horse	Pear	Tomato
Apple	100	0	0	0	0	0	0	0
Car	0	100	0	0	0	0	0	0
Cow	0	0.49	85.61	0	1.71	11.95	0	0.24
Cup	0	3.17	0.49	94.63	1.22	0.49	0	0
Dog	0	0.24	3.90	0	89.02	6.84	0	0
Horse	0	0	19.27	0	7.80	72.44	0.49	0
Pear	0.74	0	0 .24	0	0	0	99.02	0
Tomato	0.49	0	0	0	0	0	0	99.51

4.3.2 Results of Codebook Optimization

The algorithm described in section 4.2.3 is used to optimize the codebooks by removing those clusters with a very high entropy, and at the same time, retaining clusters with a moderately high entropy that are useful for classification. Note that clustering algorithms like K-means produce a different set of clusters during each run, because of the random initialization of the cluster centers. Moreover, even for slightly different set of data points, the clustering by K-means can produce very different results. This phenomenon is illustrated in Fig. 4.4 for different cases in the leave-one-object-out cross validation on the ETH-80 dataset. Figure 4.4(a) and (b) show instances where a threshold below 0.50 is ideal, however, Fig. 4.4(d) and (e) show cases where a threshold above

Table 4.5: Confusion Matrix (%) for the best result of our system on the ETH-80 database.

	Apple	Car	Cow	Cup	Dog	Horse	Pear	Tomato
Apple	100	0	0	0	0	0	0	0
Car	0	100	0	0	0	0	0	0
Cow	0	1.46	94.15	0	1.22	3.17	0	0
Cup	0	0	0	97.80	0	0	0	2.20
Dog	0	0	1.46	0	94.63	3.91	0	0
Horse	0	0	4.88	0	4.14	90.98	0	0
Pear	0	0	0	0	0	0	100	0
Tomato	0.49	0	0	0	0	0	0	99.51

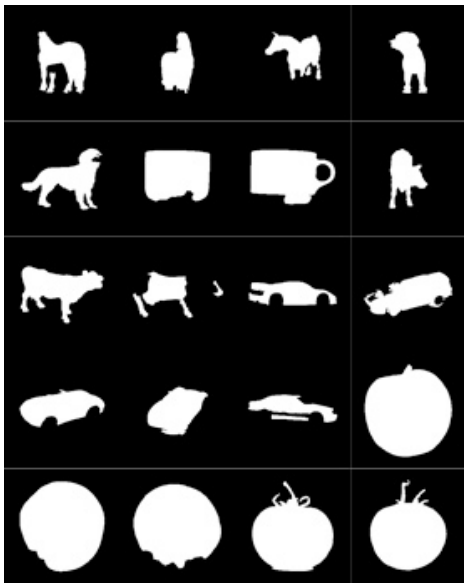


Figure 4.3: Sample shapes extracted using the salient object detection algorithm.

0.50 is optimal. Additionally, in some cases like Fig. 4.4(c) and (f), best classification can be obtained for both low and high thresholds. Therefore, in order to choose a robust threshold, cross-validation should be adopted to obtain the general trend.

Figure 4.5 shows the cross-validation accuracy as the threshold is varied from 0 to 1. As expected, the accuracy is lowest when only the "pure" clusters (low entropy) are retained. The accuracy increases as more clusters with members from different object categories (higher entropy) are introduced. Eventually,

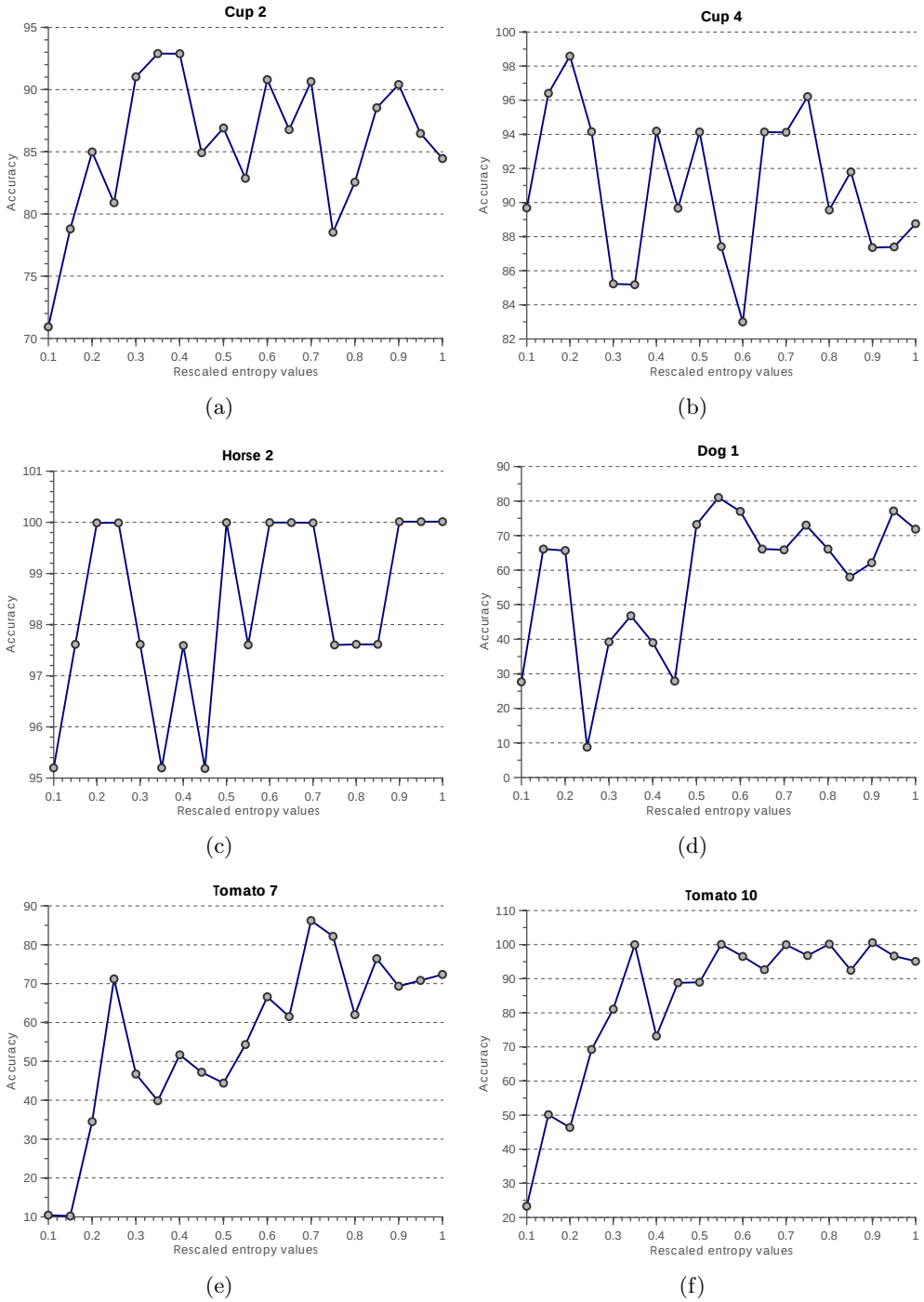


Figure 4.4: Individual cases of codebook optimization. The title of each graph shows the object that was left out for testing.

Table 4.6: Evaluation of the quality of the extracted shapes using different thresholding methods.

Thresholding algorithm	Precision	Recall	F-score
Otsu [173]	99.67	93.22	**96.34**
Kittler and Illingworth [170]	95.40	95.39	95.83
Rosenfeld and La Torre [174]	52.78	97.68	68.51
Kapur, Sahoo and Wong [175]	86.36	95.23	90.58
Prewitt and Mendelsohn [171]	99.63	92.88	96.14
Prewitt and Mendelsohn2 [171]	99.61	92.77	96.07
Glasbey [176]	96.16	95.18	95.67
Doyle [177]	46.43	97.99	63.92
Tsai [172]	99.33	92.77	95.15

the accuracy saturates for threshold values close to 1, which is the instance of using all the clusters for classification. However, when using all the clusters, the cross-validation accuracy is lower when compared to retaining clusters with 90% to 95% of the entropy energy. We note that this observation is similar to a popular way of choosing the number of principal components for dimensionality reduction: taking the first k eigenvectors that capture at least 95% of the total variance.

The best accuracy reported (97.1314%) in this thesis was obtained with a threshold of 0.90, which corresponds to selecting [2419, 2158, 2686] words for the structure, texture and shape codebooks, respectively. The original codebook size of 3000 for each cue yielded a sub-optimal accuracy of 96.2543% at a higher computational cost. After the codebook optimization procedure, a total of 1737 codewords were discarded. Therefore, there are two important advantages of optimizing the codebook.

1. Increase in classification accuracy because of removing clusters with very high entropy that potentially consist of noisy local features.

2. Since there is a significant reduction in the codebook size, computational

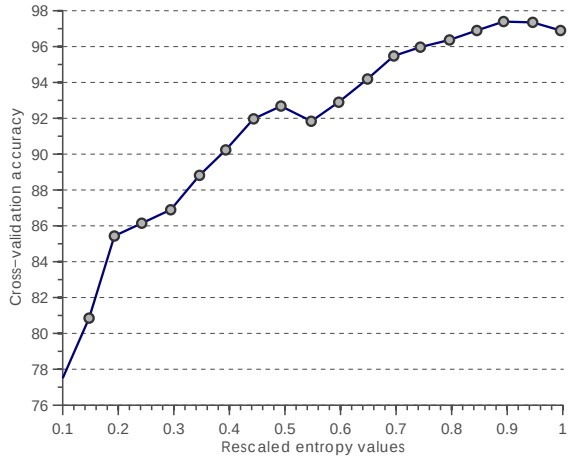


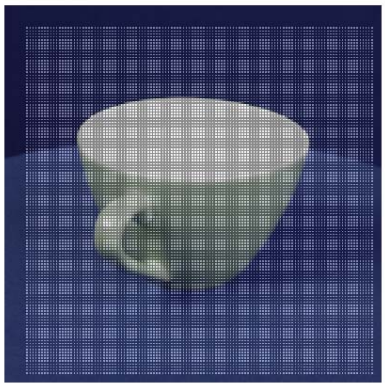
Figure 4.5: Cross-validation accuracy for various entropy thresholds.

cost for vector quantization is significantly lowered.

In the next subsection, we demonstrate the effectiveness of the proposed keypoint detection scheme.

4.3.3 Dense Sampling vs. Keypoint Detection

As far as selecting keypoints for the bag-of-words representation is concerned, the most successful method has been the dense sampling strategy. In this method, keypoints are placed all over the image without considering any explicit way to detect keypoints. It has been found that dense sampling gives equal or better classification rates than sophisticated multi-scale interest point operators. This behavior is explained by observing that the number of keypoints is the most important factor governing the performance of the bag-of-words model. It has been widely reported in the literature that more number of keypoints lead to a better performance. Naturally, keypoint detectors provide lesser sampling locations compared to a dense/random sampling strategy. However, there are a few scenarios in which dense sampling may not be suitable, namely when there



(a) Step size 2

(b) Step size 4

Figure 4.6: Two common settings for dense local sampling.

is no background context information available (plain background), or when the object size is small in comparison to the size of the picture. In both these cases, dense sampling is better avoided. This proposition is confirmed in our experiments on the ETH-80 dataset, wherein the background is uniform for all the objects and some objects like car occupy less than 20% of the total number of pixels in the image.

We implement two commonly used step sizes (two and four) for the dense

sampling method. The step size indicates the space between two keypoints in the x and y directions (Fig. 4.6). The ROF keypoint detection scheme proposed in this thesis is replaced by the dense sampling strategy for the grayscale cues while the binary shapes are still sampled at the boundaries. In comparison to the result of the proposed approach (97.13%), the classification results of the dense sampling approach are 92.9573% and 87.4695% the step size of two and four, respectively. Clearly, we can see that dense sampling produces sub-optimal results compared to a keypoint detection scheme when there is no background context information. Therefore, we conclude that it is better to use edge maps or more sophisticated methods to obtain the keypoints as demonstrated above.

4.4 Summary

In this chapter, we proposed a general framework for grayscale image classification with several key contributions. Firstly, we proposed novel local descriptors using log-polar transform for encoding structure, texture, and shape cues. Secondly, we proposed a new keypoint detection scheme using image denoising and demonstrated that it outperforms dense grid sampling by a large margin. Additionally, we proposed a codebook optimization scheme that can improve the classification accuracy while significantly reducing the codebook size. Lastly, we proposed a novel scheme to extract multiple object cues from grayscale images, and demonstrated very high classification performance on the images from the widely used ETH-80 dataset. This framework can be extended to classify color images by incorporating a separate color cue channel, which remains to be investigated.

Chapter 5

Multiple Object Cues for High Performance Vector Quantization

5.1 Introduction

After developing the grayscale image classification system in the previous chapter, we now consider the more general case of classifying objects from color images. The main idea is to integrate multiple object cues, such as structure, texture, color, and shape, using the bag-of-words model with a novel keypoint detection scheme that achieves a comparable classification accuracy to the widely-used dense keypoint strategy, at a much lower computational cost. In contrast to many works that use advanced encoding techniques or machine learning systems, we use the simple vector quantization on the proposed multi-cue representation and demonstrate highly competitive classification performance compared

to state-of-the-art algorithms on the popular Caltech-101 dataset. Additionally, we significantly outperform several state-of-the-art methods on the MICC Flickr-101 dataset, which is an updated version of Caltech-101.

The remainder of this chapter is organized as follows. We introduce the details of our proposed methods in section 5.2. Next, we evaluate the proposed framework on two popular datasets and present the experimental results in section 5.3. Finally, we conclude the chapter in section 5.4.

5.2 Multi-Cue Object Representation

We employ the bag-of-words framework consisting of four main stages: keypoint detection, feature extraction, vector quantization, and classification. For classifying the color images, we extract four object cues, namely the color, structure, texture and shape. For the appearance cues (structure and texture), keypoint detection is performed using differential entropy; for the extracted binary image, the keypoints are simply the boundary points of the shape; and for the color cue, each pixel represents a keypoint. Feature extraction involves sampling the cue images at the keypoints, using log-polar transform. The set of descriptors from each cue of the training images are collectively used to obtain a codebook. In this case, four codebooks will be generated using the training set, that is, one each for the structure, texture, shape, and color cue. Then, the quantization step is the histogram representation of the features obtained for each cue, using the respective codebooks generated in the previous step. Subsequently, the histograms of the training images are formed by a *late fusion* step, i.e., the histograms obtained for all the cues are concatenated to form the final

representation of each image. Finally, the histograms of the training images are used to train an SVM classifier. During testing, the codebook construction step is bypassed, and a test image is simply represented using the codebooks and classified using SVM. Figure 5.1 illustrates the details of the proposed cue-based feature extraction step.

5.2.1 Keypoint Detection

Similar to the previous chapter, we define the keypoints as locations on the image with a distinctive appearance with respect to its neighboring pixels. However, in contrast to the denoised edge map used in the previous chapter, we now consider a more general approach for obtaining keypoints. Therefore, to deal with noise, entropy is a useful measure to quantify the randomness of pixel values within a neighborhood. However, entropy does not take into account the degree of variance, when considering the most straightforward definition of it as an expectation:

$$H(X) = -E_X[\log_2(P(X))] = -\sum_{x_i \in \Omega_X} \log_2(P(X = x_i))P(X = x_i) \qquad (5.1)$$

where 0 $log(0)$ is defined to be 0. In the above equation, the random variable X takes discrete values in the range $[0, 255]$, corresponding to the domain of the grayscale pixel intensity values. The quantity $P(X)$ can be easily obtained by normalizing the histogram of the intensity values. For instance, a 3 by 3 neighborhood containing three 20s, three 30s, and three 33s, has a probability of $\frac{1}{3}$ each for 20, 30 and 33, and zero for the rest of the intensity values. Thus, the entropy of this distribution is $3 \times \frac{1}{3} log_2(3) = log_2(3)$ bits. Discouragingly,

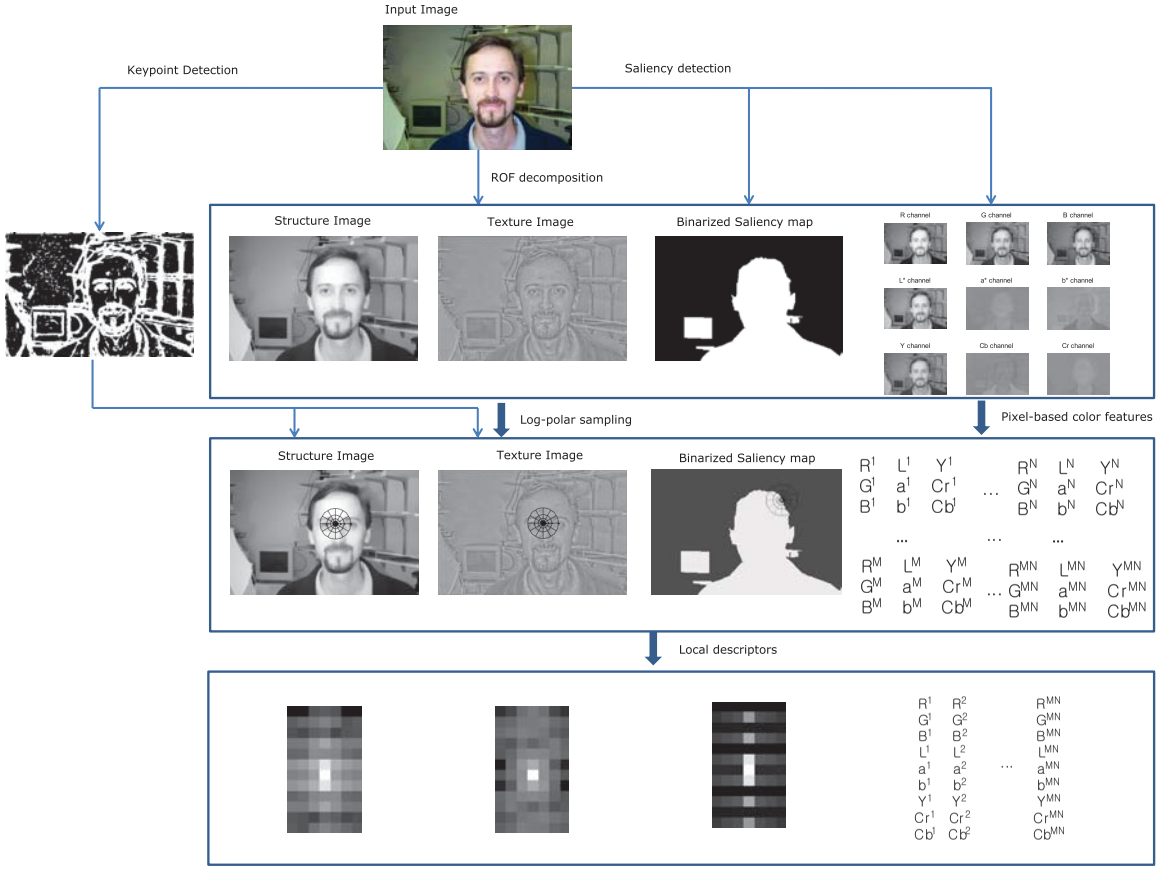
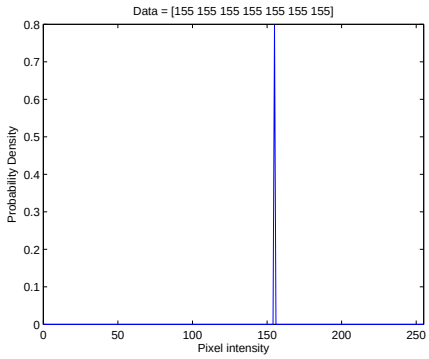
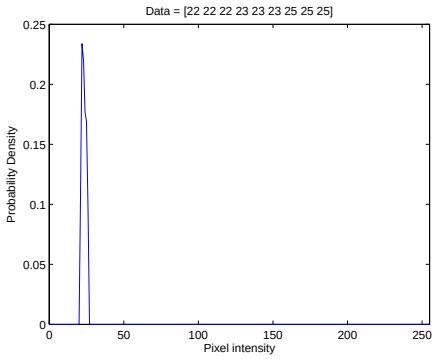


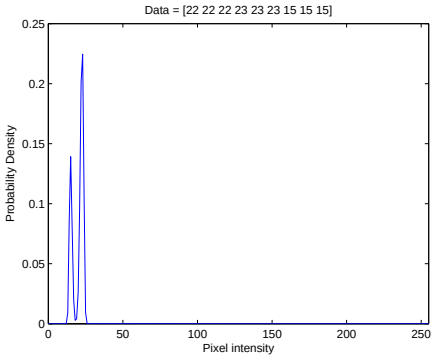
Figure 5.1: Feature extraction for cue-based object categorization (best viewed on a monitor). On the one hand, the input image is decomposed into structure and texture using the ROF method. On the other hand, saliency detection is performed on the input image to obtain a saliency map, which is further binarized using the Otsu method. Using log-polar transform, the keypoints obtained from the differential entropy map are used to sample the grayscale appearance cues (structure and texture) while the binary shape is sampled at its boundaries. In addition, local color descriptors are obtained by using the pixel values in different channels of the RGB, CIE LAB and YCbCr color spaces.



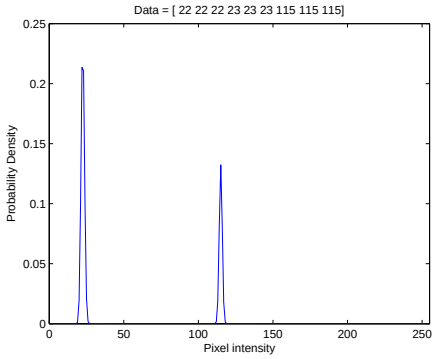
(a) Differential entropy = 0.1802



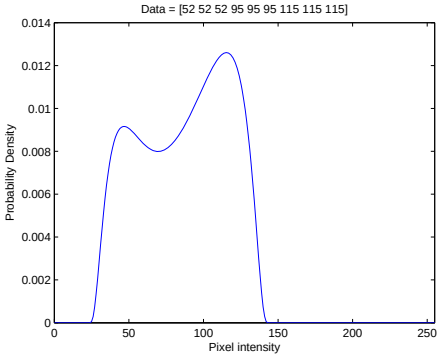
(b) Differential entropy = 1.7485



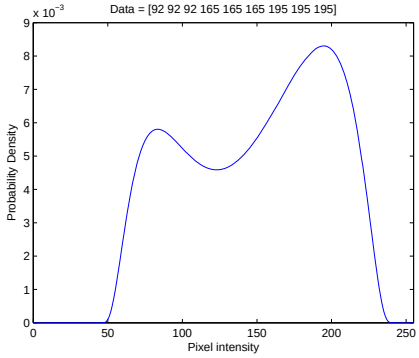
(c) Differential entropy = 2.1008



(d) Differential entropy = 2.1354



(e) Differential entropy = 4.6776



(f) Differential entropy = 5.1439

Figure 5.2: Differential entropy for various pixel intensity values in a neighborhood. For all the above cases except (a), the discrete entropy is the same, which is $log_2(3)$ bits. For case (a), discrete entropy is zero, whereas differential entropy is non-zero.

the entropy of a very dissimilar neighborhood is also the same as long as the probability distribution remains the same. For instance, a 3 by 3 neighborhood containing three 80s, three 255s, and three 0s, also has an entropy of $log_2(3)$ bits. This point is illustrated in Fig. 5.2 with a few examples and it is shown why the continuous version of entropy is more useful for quantifying the variance of data while having a relative measure of its randomness.

The continuous version of entropy is known as differential entropy, and it is defined as follows:

$$h(X) = -E_X[\log_2(p(X))] = -\int_{-\infty}^{\infty} p(x)\log_2(p(x))dx \tag{5.2}$$

Now the problem is to estimate the probability density function (pdf), $p(x) : x \in \Omega_X$. To this end, we use kernel density estimation [178, 179] to obtain the pdf of the pixel values in a pre-defined 5 by 5 neighborhood at each keypoint. Kernel density estimation is the most popular nonparametric approach to density estimation because of its flexibility in modeling a given dataset while being unaffected by the bias of specifying a particular model [180].

Let $(x_1, x_2, \cdots, x_n)$ be an independent and identically distributed sample drawn from a distribution with an unknown density *f*. The kernel density estimator computes the shape of this function by using a non-negative kernel $K(\cdot)$ that integrates to one and has zero mean, and a non-negative smoothing parameter h (bandwidth), as given below.

$$\hat{f}(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{nh}\sum_{i=1}^{n} K(\frac{x - x_i}{h}) \tag{5.3}$$

The above equation can be used to interpret the histogram by considering a rectangular kernel of area one (the width and height determine the bin size) and obtaining the estimate of the pdf at a given point as $1/n$ times the sum of the heights of all the rectangles that cover the point. Instead of using rectangles, a range of weighting functions are commonly used for the kernel density estimate: triangular, Gaussian, Epanechnikov [181], and others. For the listed kernels, the loss of efficiency is comparable [182], and the Gaussian kernel is often used due to its convenient mathematical properties.

$$K(\frac{x - x_i}{h}) = \frac{1}{\sqrt{2\pi}} \exp\left(\frac{-(x - x_i)^2}{2h^2}\right) \tag{5.4}$$

In this work, we adopt the Gaussian kernel with appropriate selection of bandwidth using the method proposed in [183], which is also known as the Silverman's rule of thumb for Gaussian basis functions. The choice of h is given by

$$h = \left(\frac{4\hat{\sigma}^5}{3n}\right)^{\frac{1}{5}} \tag{5.5}$$

where $\hat{\sigma}$ is the standard deviation of the samples. Once the pdf is estimated, differential entropy can be found using equation 5.2. As seen from Fig. 5.2, samples with numerically close intensity values have a lower differential entropy compared to those with spread-out intensity values. Thus, when the differential entropy map used to select the keypoints, those pixels with a dissimilar neighborhood will be selected as keypoints. This method to select keypoints is consistent with the idea of "surprise" in saliency and related psychophysical literature [89, 98, 99, 184, 185], i.e., those regions that are very different from its

surroundings bring about the attention of the human vision system. To further increase the contrast of the differential entropy map, histogram equalization [186] is carried out before thresholding it using the Otsu method to obtain the keypoints.

5.2.2 Feature Extraction

Except for the keypoints selection step described above, the procedure to obtain the local features for the grayscale structure, texture and shape cue remain the same as presented in the previous chapter. For the color cue, the descriptors are obtained using the pixel values in different color spaces without considering keypoint selection. Additionally, color GIST descriptors [187] are extracted to obtain a low-dimensional representation of the image.

5.2.3 Contextual Information using Cooccurrence Signature

It has been observed that very high codebook sizes (> 10000) are crucial for the better performance of vector quantization, which has been used throughout this thesis. However, advanced image encoding techniques, such as Fisher vector encoding, utilize less than 500 codewords while still managing to outperform the vector quantization approach. The reason being that advanced image encoding techniques extract more detailed statistics between the codewords and the local features (contextual information), and thus fare better than their simpler counterpart. Taking inspiration from these encoding approaches, we aim to improve the bag-of-words model by using small codebooks to encode contextual information.

As pointed out in Chapter 3, there are many ways to encode contextual in-

formation in the bag-of-words model. One method of choice is to derive features from the spatial co-occurrence matrix [58]. In general, co-occurrence features are obtained by thresholding the co-occurrence matrix [60], or extracting class-unique bigrams as proposed in Chapter 3, or extracting Markov stationary features, etc. All these approaches aim to extract a subset of the information from the co-occurrence matrix due to its large size. For instance, a commonly used codebook size of 1024, yields a spatial co-occurrence matrix with 1024×1024 elements, which cannot be used directly even on modern PCs for training the SVM. In this chapter, we propose to use a small codebook size of N (of the order of 100) and obtain the contextual representation with $N(N+1)/2$ features.

The spatial co-occurrence matrix is created by recording the total number of times a pair of neighboring local descriptors gets assigned to c_i and c_j, which are any two of the N visual words denoted as $S = \{c_1, c_2, ...c_N\}$. Therefore, the size of the co-occurrence matrix is $\mathbf{C} \in \mathrm{R}^{\mathrm{NxN}}$, in which each entry $\mathbf{C}(i,j)$ is computed by inspecting every keypoint and its immediate neighboring keypoints (3 by 3 neighbourhood) to count the number of times the neighboring descriptors form an $i-j$ pair. Since the co-occurrence matrix is symmetrical, we use the upper triangular part to form the histogram of bi-grams without any mining operation to extract discriminative bi-grams. The reason for this strategy lies in the low codebook size, i.e., since the codewords are not very specific to the image content, they are expected to form strong associations with a few object categories, but at the same time, not create broad associations with all the categories. Therefore, each $\mathbf{C}(i,j)$ entry becomes potentially discriminative if used along with all the non-redundant ones to form a unique "signature" for each object category. We

term this simple yet powerful method to obtain the histogram of bi-grams as "co-occurrence signature".

5.2.4 Vector Quantization and Classification

For the grayscale appearance and binary shape cues, a training/testing image is quantized into K_i histogram bins ($i = 1, 2, \text{and } 3$ for structure, texture and shape respectively), i.e., the local features extracted from a cue image are individually matched to the nearest visual word of the respective codebook using Euclidean distance and the frequency of each word creates the K_i-dimensional histogram representation. Besides the global bag-of-words features, 2×2 and 3×3 image grids are used to capture mid-level spatial information. Each of the histograms from the grid regions is normalized separately and concatenated together. In turn, the $13K_i$-dimensional vector using the image grids is concatenated with the K_i-dimensional vector to form the $14K_i$-dimensional cue representation. For the binary shape cue, the normalized histogram of bi-grams is further concatenated to produce a vector of dimension - $(14K_3 + N_{bi})$, where $N_{bi} = N(N+1)/2$. Typically, the binary shape has a few thousand keypoints, whereas the other cues have in the order of a few ten thousand keypoints. Therefore, we only encode bi-grams for the shape cue as encoding it for the other cues give slightly less performance, because of the use of small codebooks to encode a large number of descriptors.

For the global representation of the color cue, we used the color GIST descriptor instead of the global bag-of-words representation, which did not increase or decrease the performance significantly, possibly because the pixel based color features are not useful without spatial information. The GIST descriptor is ob-

tained for each of the color channels in RGB, YCrCb and La*b* spaces and they are concatenated to form the color GIST descriptor (512 x 9 = 4608 dimensions). Apart from the global bag-of-words representation, encoding the mid-level information for the color cue remains the same as explained earlier (denoted as $13K_4$). Therefore, the color cue representation is of dimension $13K_4 + 4608$. The final image representation is formed by concatenating the four cue representations to produce a $(14K_1 + 14K_2 + 14K_3 + N_{bi} + 13K_4 + 4608)$-dimensional vector. The classifier used is the SVM implementation of VLFeat in their bag-of-words application [152].

5.3 Experiments and Discussion

We tested our object classification system on two standard datasets, the widely tested Caltech-101 [188] and its updated version Flickr-101 object dataset [189]. The Caltech-101 object dataset consists of 101 object categories with varied number of images in each category (minimum of 31 and a maximum of 800). The experimental protocol of this dataset is to train on 30 images and test with a maximum of 50 images per category. Most images in Caltech-101 have the object centered in the image with little or no clutter in a stereotypical pose. Therefore, MICC-Flickr101 was conceived with the idea of cloning Caltech 101, but with realistic representations collected from the internet photo sharing website, Flickr. The experimental protocol of Flickr-101 is the same as its predecessor, however, the images are significantly more complex and challenging than Caltech101 (see Fig. 5.3). Our experiments were carried out on HP Xeon Two Sockets Quad-Core 64-bit Linux clusters with 72 GB memory limit.

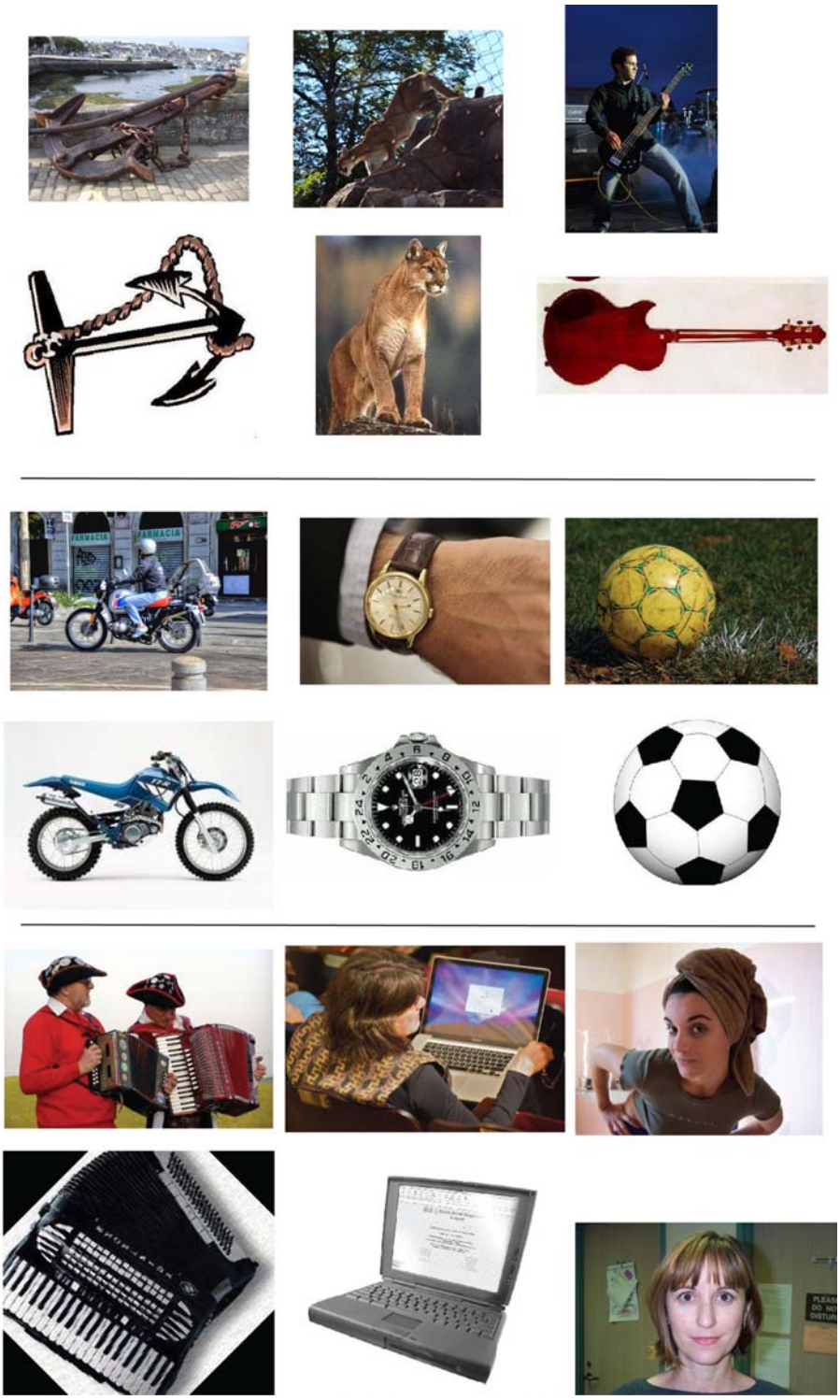


Figure 5.3: Sample images from Caltech-101 (even rows) and Flickr-101 (odd rows) datasets. Anchor, cougar body, electric guitar, motorcycle, watch, soccer ball, accordion, laptop, and faces, are the objects named from left to right in each set of row.

Table 5.1: Classification accuracy comparison of the proposed method with previous works on Caltech-101 dataset (%).

Method	Accuracy
Gemert [190]	64.16
SVM-KNN [191]	66.20
SPM [50]	64.60
Griffin [192]	67.60
Boiman [193]	70.40
Jain [194]	69.10
Yang [52]	73.20
LLC [91]	73.44
Jia [53]	75.30
SLRR [195]	73.60
LSGC [196]	75.10
RBC [94]	75.60
Chatfield [95]	77.78
MKL [197]	77.20
BCF [76]	77.80
Gehler [69]	77.80
Kanan [198]	78.50
SSC [199]	79.84
HLM [200]	79.63
P-SIFT [201]	80.13
Proposed Method	**80.15**

5.3.1 Classification Results on Caltech-101

We compare the performance of the proposed method to many seminal works on the Caltech-101 dataset in Table 5.1. It is worth mentioning that with the use of vector quantization alone, our cue-based bag-of-words framework outperforms the previous works by a comfortable margin. In other words, we did not make use of successful encoding methods like LLC [91], RBC [94], sparse coding [52, 53], etc., or multiple kernels [197] to boost the classification accuracy. On the other hand, some earlier works like [69, 193] do use multiple feature descriptors like the proposed features in this chapter. For instance, a combination of SIFT, luminance descriptor, color descriptor, shape descriptor, and the self-similarity descriptor was used in [193]. Note that irrespective of the features used by

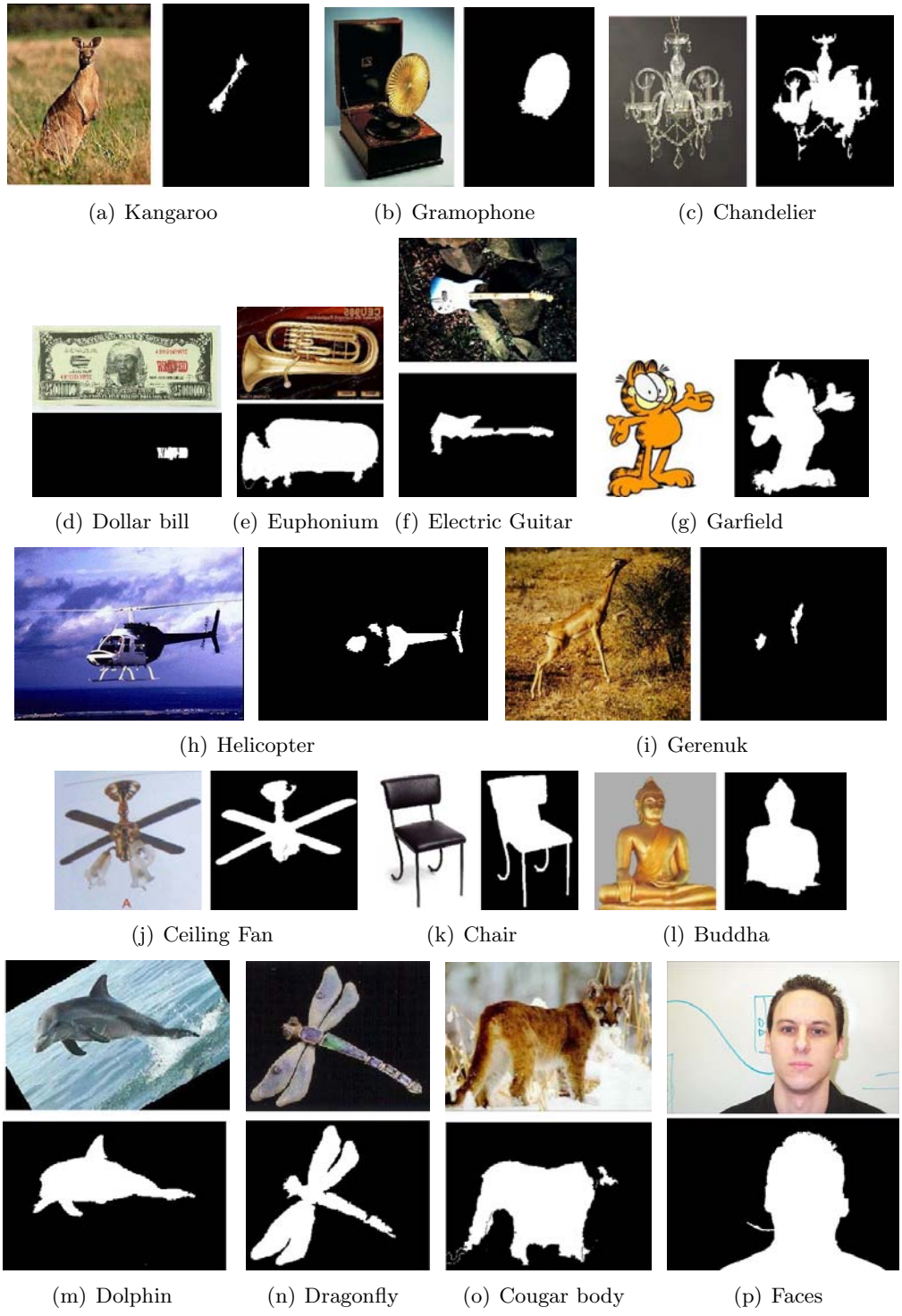
Table 5.2: Performance of individual object cues in comparison with the proposed method on Caltech-101 (%).

Alg.	Str	Tex	StrTex	Shape	Bigr	Color	ColGIST	Proposed
Acc.	59.25	59.42	66.13	52.72	55.72	47.13	66.43	80.15

the previous methods, all of them follow the same experimental protocol, and hence, the comparison of the classification accuracies is valid. To the best of our knowledge, the results presented in this work are unprecedented considering the use of the bag-of-words model with vector quantization.

We are aware that deep learning methods [202] can obtain a classification accuracy around 86%, but they usually require a large amount of training data. Reference [202] used convolutional neural nets trained on the ImageNet 2012 training set (1.3 million images, spread over 1000 different classes) and tested on Caltech-101 to obtain very high performance, whereas their convnet model trained and tested using the images of Caltech-101 achieved a dismal 46.5% classification rate. Next, we show the individual performance of the object cues, and further show the necessity of multiple cues for high classification performance.

Table 5.2 shows the performance of the individual object cues in comparison with the performance of the proposed multiple cue representation. Clearly, using any individual cue leads to a poorer performance in comparison to the use of multiple cues. Moreover, only when the structure and texture cues are combined together, the performance is as good as using the colorGIST cue, which captures the overall detail of the object image using Gabor filters. Since the images of Caltech-101 are more complex than the images from a no-background dataset like ETH-80, the salient object detection algorithm do not efficiently locate the object, which leads to a poor segmentation as seen from some of the sample



(a) Kangaroo (b) Gramophone (c) Chandelier

(d) Dollar bill (e) Euphonium (f) Electric Guitar (g) Garfield

(h) Helicopter (i) Gerenuk

(j) Ceiling Fan (k) Chair (l) Buddha

(m) Dolphin (n) Dragonfly (o) Cougar body (p) Faces

Figure 5.4: Sample shapes extracted from the images of Caltech-101.

shapes extracted by thresholding the saliency maps using the Otsu method (Fig. 5.4). Overall, the proposed method of combining all four cues using four separate codebooks in the bag-of-words model gives the best result. Similar to the previous chapter, a codebook size of 2500 was initially set for all the cues, and with an entropy threshold of 0.95, the codebooks were pruned to have [1324, 887, 1479, 2072] words for the structure, texture, shape and color cue respectively. Compared to using the original codebooks, the pruned codebooks give a better classification accuracy (around 1.5% more) and also result in a faster vector quantization.

Since the confusion matrix of 101 categories is impractical to list, we note some of the best and worst performing categories of our classification system. There were 12 classes that were perfectly categorized, namely Faces (50), airplanes (50), binocular (2), car side (50), metronome (1), minaret (44), octopus (2), pagoda (17), scissors (4), snoopy (1), strawberry (5), and wild cat (3), where the number in the bracket denotes the number of test images. Categories like faces, airplanes and side view of cars have been noted as easy to classify in previous works too. There were a few classes with more than 40% of the test images classified wrongly: 'beaver' (10/15), 'cougar body' (11/17), 'crab' (19/40), 'crocodile' (11/20), 'crocodile head' (13/21), 'cup' (13/24), 'ketch' (31/50), 'scorpion' (23/50), where the numbers in the brackets indicate the number of wrongly classified images and the total number of test images. With the exception of ketch and cup, it is interesting to note that non-rigid objects like animals are difficult to classify, since the intra-class pose and appearance variations are very high. Therefore, a classification system that can leverage the benefits of many

object cues will perform much better than those using them separately, as seen from the results in Table 5.2. Next, we present experimental results on the shape classification system, which is an important component of the proposed framework.

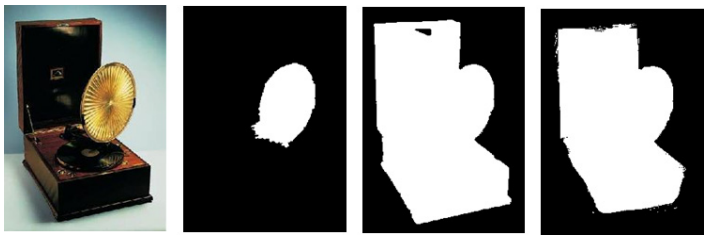
Comparison of Different Salient Object Detection Algorithms

Since there is no ground truth segmentation provided in the dataset, it is difficult to quantify the quality of the extracted shapes. The quality of the extracted shapes mainly depends upon the saliency algorithm, i.e., as long as the pixels that belong to the object have a high saliency, the thresholding method is likely to locate it, and many thresholding methods like the Otsu method have been shown to perform well for bimodal histograms [176]. Therefore, we compare different state-of-the-art salient methods [78, 203–206] using the Otsu method in terms of the shape classification accuracy. Table 5.3 lists the classification accuracy of the shape classification system when different salient object detection models are used to obtain the shape image. Note that the maximum accuracy of 47.54% was obtained by Jiang [206], which is a saliency detection method based on absorbing Markov chain. As these different saliency models take a different approach to obtain the saliency map, we found that the shapes obtained can be different for the same image and they can complement each other in many scenarios.

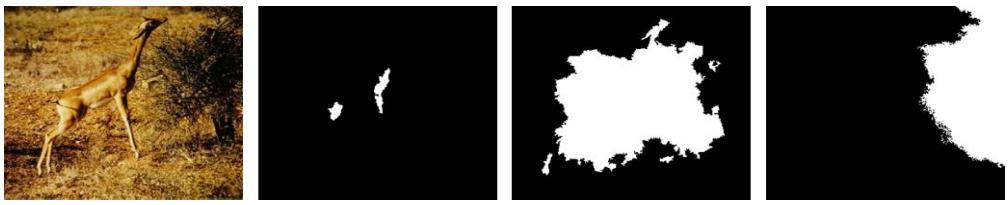
Figure 5.5 shows some shape images obtained using the three best salient object detection models, Center prior [78], Absorbing Markov Chain [206], and Multi scale superpixels [203]. It can be clearly seen that sometimes shape extraction can be difficult for all the three saliency models (Fig. 5.5(d)), or a couple of

(a) Kangaroo

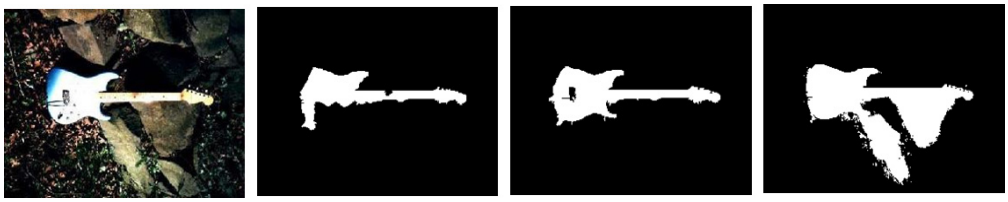
(b) Helicopter



(c) Gramophone



(d) Gerenuk



(e) Electric Guitar

(f) Cougar Body

Figure 5.5: Sample shapes extracted from different salient object detection models.

Table 5.3: Comparison of different salient object detection algorithms in terms of shape classification accuracy (%).

Saliency model	Accuracy
Multi scale superpixels [203]	39.95
Center prior [78]	38.74
Bayesian model [204]	37.10
Dense & Sparse Representation [205]	32.84
Absorbing Markov Chain [206]	47.54

Table 5.4: Classification accuracy comparison of the proposed method with previous works on Flickr-101 dataset (%).

Method	Accuracy
GIST [189]	26.10
SIFT_{bow} [189]	31.20
rgbSIFT_{bow} [189]	34.40
MKL (GIST-SIFT-rgbSIFT) [189]	39.30
Proposed Method	**49.59**

them perform well (Fig. 5.5(c) and (e)), or one of them is better (Figure 5.5 (a) and (b)), or all three perform similarly (Fig. 5.5 (f)). Since it is an extremely difficult task to predict which one of the saliency models may perform well on a given image, especially without the ground truth segmentation, we obtained all the three shapes and extracted local features using log-polar transform. Using all three shape images is implementation-wise equivalent to the common idea of obtaining multiple descriptors at different scales for the same keypoint. The classification results reported in Table 5.2 was the multiple shapes setting as described above.

5.3.2 Classification Results on Flickr-101

In Table 5.4, we compare the performance of the proposed method to many single feature settings and a multiple kernel fusion method on the Flickr-101 dataset. We outperform these methods by a big margin with the use of vector

Table 5.5: Performance of individual object cues in comparison with the proposed method on Flickr-101 (%).

Alg.	Str	Tex	StrTex	Shape	Bigr	Color	ColGIST	Proposed
Acc.	33.49	37.04	39.96	22.62	23.70	26.12	33.30	49.59

quantization on the powerful set of features proposed in this work. Table 5.5 shows the performance of individual cues and the multiple object cues setting on the Flickr-101 dataset. Again, the shape cue performs better with the bigrams, but still has lower performance compared to other cues, which leaves considerable room for improvement in the future. Note that a codebook size of 2500 was initially set for all the four codebooks, and with an entropy threshold of 0.95, the codebooks were pruned to have [1243, 1080, 1446, 1899] words for the structure, texture, shape and color cue respectively.

5.3.3 Differential Entropy Keypoints vs. Dense Sampling Strategy

The number of keypoints obtained using the proposed differential entropy approach (avg. 48% of the image pixels) is much lesser than the dense sampling strategy (avg. 85% of the image pixels). With regards to the classification accuracy, the maximum accuracy obtained by the dense sampling strategy was 79.89% while the differential entropy keypoints obtained a similar accuracy of 80.15% on the Caltech-101 dataset. On the Flickr-101 dataset, the dense sampling strategy achieved an accuracy of 49.14% while the proposed differential entropy keypoint detection method achieved a par accuracy of 49.59% at a much lesser computational cost. Moreover, with a smaller step size (two and four) for the dense sampling method, the classification accuracy dropped to 78.75%

and 75.45% respectively on the Caltech-101 dataset, which confirmed the effectiveness of the proposed keypoint detection method in terms of classification accuracy, memory requirements and computational load.

5.4 Summary

In this chapter, we proposed a general object classification framework that encodes different object cues using local descriptors obtained using the log-polar transform. Besides the framework, we introduced a novel keypoint detection method that was found to be better than the dense sampling strategy from a practical point of view, i.e., the number of local descriptors encoded is much lesser without a significant drop in accuracy. Thus, the proposed keypoint detection scheme using differential entropy offers a more principled approach to image sampling for the popular bag-of-words framework. Additionally, we also proposed a new way to encode contextual information in the bag-of-words that improves the overall accuracy without affecting the dimensionality of the features in a significant way. Using the proposed features in combination with the simple vector quantization method, we outperformed many seminal works on the widely tested Caltech-101 dataset and its recently upgraded version, the Flickr-101 dataset. Note that we compared our work to several works that used advanced encoding techniques, more powerful machine learning paradigms like the multiple kernel fusion, and advanced feature pooling techniques. Therefore, we conclude that the proposed features open up exciting possibilities for more advanced image encoding techniques.

Chapter 6

Biologically Inspired Composite Vision System for Traffic Monitoring

6.1 Introduction

After developing the object classification framework for the most general case of color images, we now consider a practical application of the log-polar transform, which was used to derive the features in the previous chapters, to video processing. The key idea is to use log-polar transform to stitch video information acquired from cameras of different visual field depths into a single video stream, and thereby, track moving objects in the log-polar space with a much longer tracking range compared to using a single camera. This composite vision system is applied to the problem of real-world speed estimation and license plate detection of vehicles in expressways.

The rest of this chapter is organized as follows. Firstly, an overview of the proposed traffic monitoring system is given in section 6.2. Next, section 6.3 presents the composite camera design with implementation details. Then, section 6.4 presents the vehicle tracking algorithm along with the proposed vehicle speed calculation algorithm, followed by section 6.5 which provides details about the license plate detection system. Experimental results with discussion are given in section 6.6. Finally, conclusions and future works are presented in section 6.7.

6.2 Overview of the Traffic Monitoring System

Figure 6.1 shows the proposed traffic monitoring system with simultaneous near and far field viewing capability for monitoring vehicle movements up to 1000 m away from the shooting point. The proposed system tracks the moving vehicles while they are present in the composite camera's field of view. Using the tracking information, the speed of each vehicle is estimated and whenever there is an instance of overspeeding, a third camera is triggered to output the license plate information of the overspeeding vehicle, which can be easily used for law enforcement. As mentioned in Chapter 1, the drawbacks of traditional RADAR traffic monitoring systems are overcome by adopting a vision-based approach to traffic monitoring. Additionally, the proposed system overcomes the problems faced by existing vision-based traffic monitoring systems by providing a longer tracking range (up to 3 times longer), and also offers new insights into simultaneous far and near field imaging.

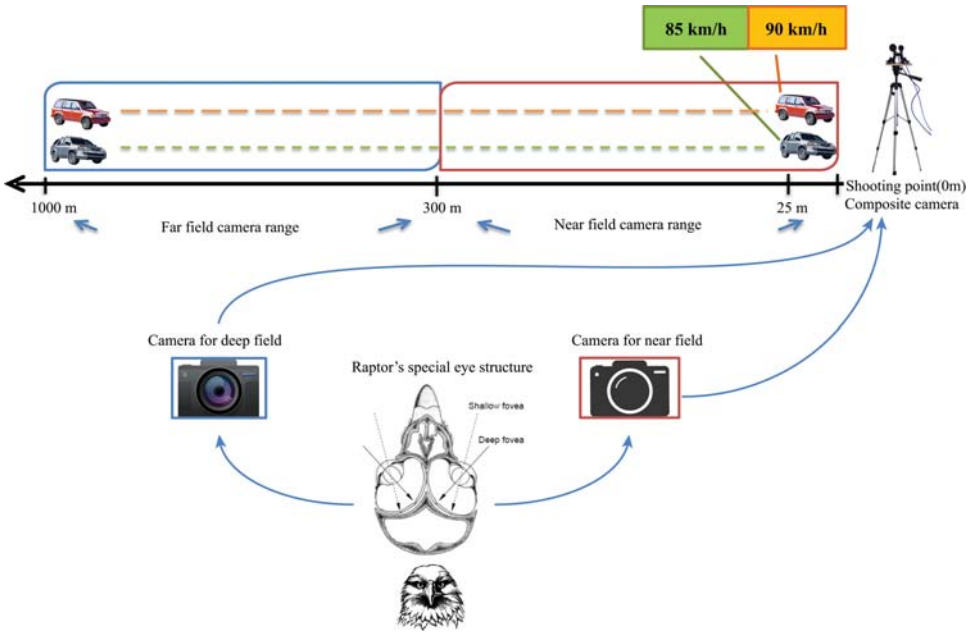


Figure 6.1: Overview of the Composite Vision System.

6.3 Composite Camera Design

Some birds of prey have the ability to visually focus on objects in both near and far field simultaneously [207, 208], which explains why they are able to perceive their surroundings to avoid hazards while being able to target a prey at a long distance. The bottom portion of Figure 6.1 shows the structure of a raptors eye with two sets of fovea. The shallow field fovea is used for navigation purposes whereas the deep field fovea is used for locating the far away prey. These two sets of fovea are simulated using two cameras with different depth-of-fields. Since the study about the raptor's foveae to brain mapping is limited, we adopt the primate retina model (log-polar mapping [32]) to simulate the raptor's internal mapping.

Using log-polar mapping, the vision information from each camera can be transformed into the log-polar space regardless of the depth-of-field. Subsequently, the problem of combining information from multiple depth-of-fields is

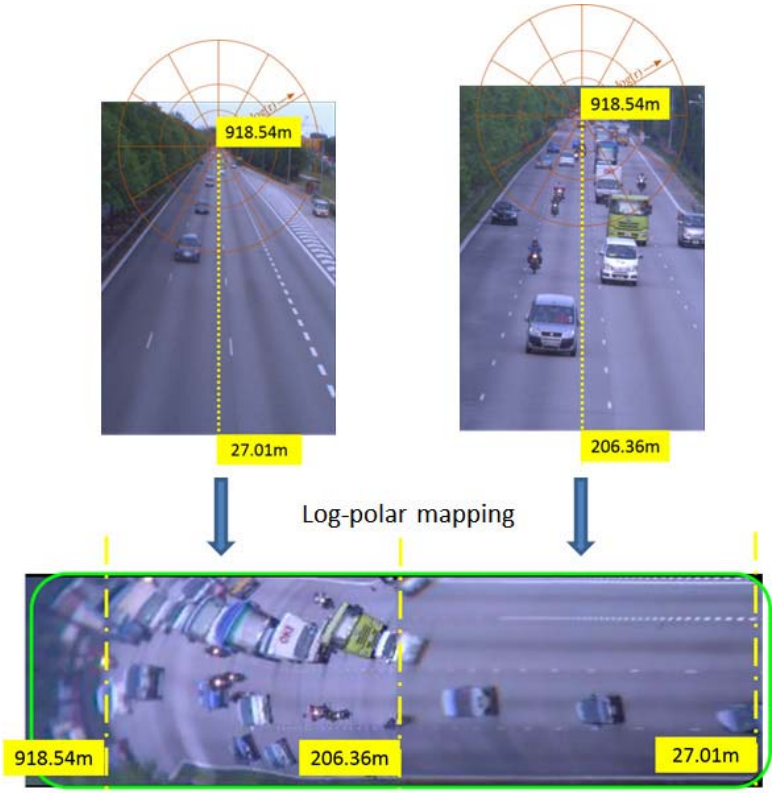


Figure 6.2: Composite image stitching example. Best viewed in color.

neatly solved by simply concatenating the log-polar encoded images, as shown in Figure 6.2. Furthermore, log-polar transformation (LPT) with ideal center point provides scale and rotation invariance. As a result, the scale change of vehicles caused by forward vehicle movement will be converted to horizontal shifts in the log-polar space with a fixed shape. Hence, the transformed LPT image could possibly provide relatively unchanged vehicle shape during the tracking process.

In order to form a single video stream for tracking purposes, the following steps are carried out on each video frame from the two cameras that view the scene synchronously. The vanishing point of the images from each camera is selected to be the corresponding center of the log polar transformation. Note that the vanishing point selection needs to be done only once using the first frame of each video. Subsequently, log polar mapping is carried out on each

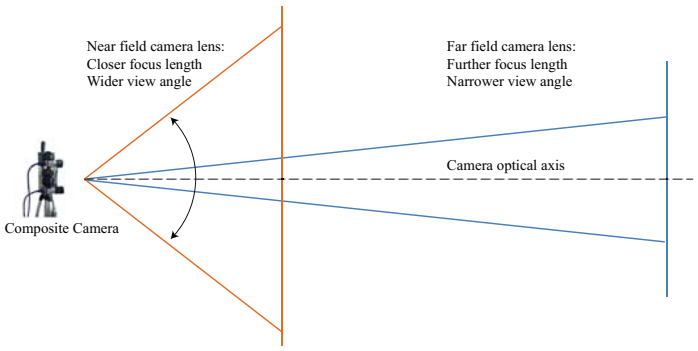


Figure 6.3: Relationship between the individual cameras in the composite camera setup.

frame, and redundant image information is cropped out to retain only the area of interest. A simple concatenation of the LPT frames is then carried out to form a single video containing near and far field information.

In order to have seamless stitching, the composite camera requires two cameras with different view angles and foci. In addition, the two cameras should be placed as close as possible to reduce errors. As shown in Figure 6.3, these nested cameras should have a special relation to achieve seamless stitching of different depth-of-field information. For ideal log-polar mapping and stitching, the camera relation factor K_b (equation 6.1) is set to be 10. In other words, the two camera lenses should have roughly about 10 times difference in their view angle.

$$K_b = \frac{tan(\theta_{far}/2)}{tan(\theta_{near}/2)} \tag{6.1}$$

$$K_b \approx \frac{\theta_{far}}{\theta_{near}}, if \theta_{far} \to 0, \theta_{near} \to 0 \tag{6.2}$$

where θ_{far} is the viewing angle of the far-field camera, θ_{near} is the viewing angle of the near-field camera,

	FireWire	Gigabit Ethernet	USB 2.0	USB 3.0	Camera Link
Bandwidth	80MB/s	100MB/s	40MB/s	440MB/s	680MB/s
Cable length	10m	100m	5m	3m	7m
Consumer acceptance	Declining	Excellent	Excellent	Excellent	None
Multiple cameras	Excellent	Good	Fair	Excellent	Fair
Power delivery	Excellent	Excellent (POE)	Fair	Good	None
Vision Standard	IIDC DCAM	GigE Vision	No	USB3 Vision	Camera Link

Figure 6.4: Industrial Camera Standards.

(a) Basler cameras. (b) Composite camera system.

Figure 6.5: Composite Camera built using USB 3.0 industrial cameras.

6.3.1 Composite Camera Implementation

For the camera design introduced earlier, a survey in the market revealed that there is no readily available device for synchronous viewing with different depth-of-fields. Therefore, the composite camera was implemented by choosing individual cameras and integrating them in a flexible hardware mount. After a thorough consideration, USB 3.0 standard cameras were chosen for implementation due to their plug-and-play usability (no extra power supply needed) and high bandwidth capability. These properties make USB 3.0 standards one of the best, as highlighted in Figure 6.4 [209]. Further review about industrial USB 3.0 cameras led us to choose Basler cameras, shown in Figure 6.5(a).

Two lenses were carefully selected for the implementation of the composite

camera to simulate the vision of raptors. One of them is a wide angle camera which provides a 76.7 degree view angle, and the other has a narrow view angle about 7.9 degrees. Therefore, the two of them have approximately 10 times difference in viewing angle. Figure 6.5(b) shows the composite camera setup, which includes a flexible hardware mount designed to adjust the cameras' positional relationship arbitrarily. The next step is to synchronously capture videos from both these cameras using a stable software that avoids frame rate drops.

Random frame rate drops of either or both cameras in the composite setup will cause the two videos to be out of synchronization. Consequently, the LPT-stitched video will have various tracking issues like cars suddenly vanishing or double images of them at the stitch line. To prevent frame rate dropping issues, all frames acquired by the cameras are first stored into a buffer before writing it to a single video file. To make sure that the buffer operation is fast enough, it is performed in the system RAM since read/write operation in RAM is generally faster than that of in a hard-disk. However, the buffer can become saturated with the image data and cause an abrupt crash of the program. Hence, a multi-threading program was implemented to split the individual operations and carry them out concurrently to save memory and processing time.

Our initial experiments to synchronously capture videos using MATLAB was unsuccessful, because it could not differentiate the identical cameras with different lenses. Hence, a program was written in C++ making use of the Application Programming Interface (API) provided by the camera manufacturer. Thus, camera settings, such as frame rate, shutter speed, and exposure time, were configured to be used by the video acquisition program. This program can

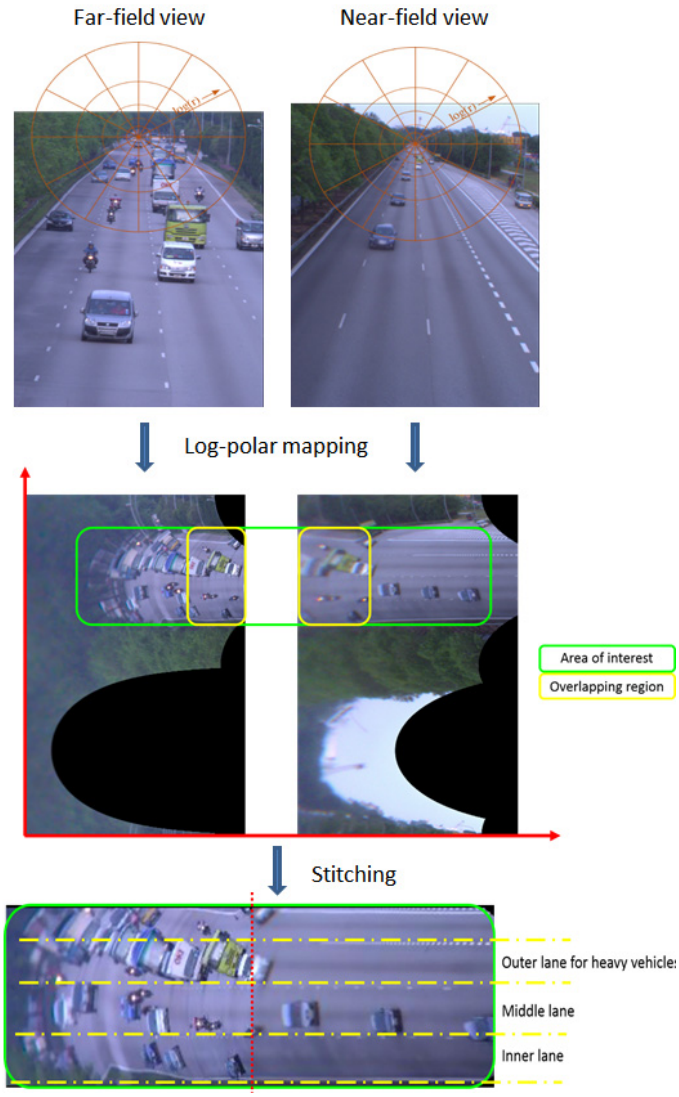


Figure 6.6: A typical stitching process of the multiple depth-of-field images.

accommodate multiple cameras for synchronous recording provided that they are all connected to USB 3.0 ports.

6.3.2 Multiple Depth-of-Field Data Processing

The detailed stitching procedure of the video frames acquired from the composite camera is shown in Figure 6.6. For any two Cartesian video frames acquired synchronously from the composite camera, log-polar transformation is applied individually. To achieve scale and rotation invariance, the road vanish-

ing point has to be carefully determined, which serves as the center point for the log-polar transformation. After the transformation, there appears an over-sampled region around the inner part of the near view and an under-sampled region around the outer part of the far view, shown in Figure 6.6 as the blue areas. To stitch the two views, the duplicated regions in both views are discarded. The final stitched result has well-separated road lanes that assist tracking in particular lane(s) of interest. Notice that the stitched result has a significant reduction in image content, which helps to speed up the object tracking process and increase the computational efficiency. As shown in Figure 6.6, the Cartesian image size of both camera views is 600 x 900, whereas the stitched log-polar space composite view result is 136 x 509 only. Moreover, the composite camera video extends the object tracking range from about 300 m in the traditional vision-based methods to up to 1000 m.

6.4 Vehicle Tracking and Speed Estimation

To detect objects in motion, the background subtraction approach works efficiently when the camera is stationary, which is conveniently the case in our work. Furthermore, to counter the complicated outdoor conditions, such as landscape changes due to shadows of clouds/trees/vehicles, low lighting conditions and reflections off the vehicle chassis, Gaussian mixture model (GMM) [210] is adopted to extract and separate the moving vehicles from the background. The system compares the video frames to a background model [211] to determine whether individual pixels are part of the background or the foreground. With sufficient training frames (150) and a small enough learning rate (0.05), it can distinguish

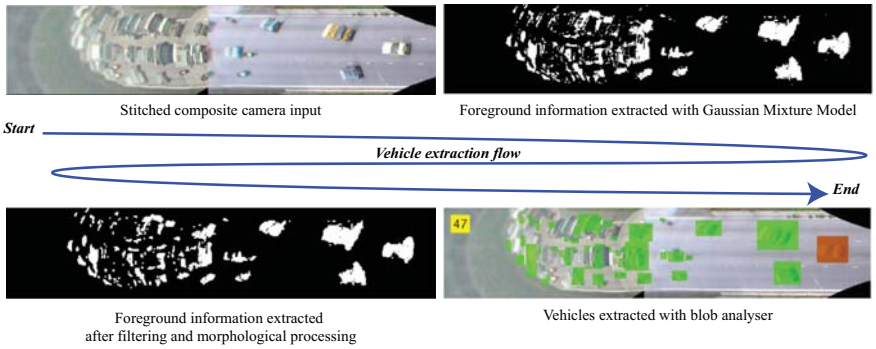


Figure 6.7: Vehicle extraction with Gaussian mixture model.

between moving objects and the background, even if the environment is contaminated with noise or illumination variations. For details about the real-time tracker implementation, the reader is referred to [210, 211]. To further remove noisy detections, morphological area opening[1] is performed to remove blobs with less than 80 pixels. Subsequently, blob analysis yields the bounding boxes of the foreground vehicles with their centroid locations. Figure 6.7 shows the vehicle extraction result using the GMM algorithm.

After extracting the vehicles using GMM, the next step is to track them in the midst of complex motion, such as lane switching, sudden acceleration and deceleration, etc. To achieve this goal, Kalman filter [212] was adopted for simultaneously tracking multiple vehicles in the scene. The main advantage of using Kalman filter is its ability to model the vehicle's acceleration in the video due to the prospective projection. In addition, it provides tolerance to a certain degree of occlusion by predicting the vehicle position based on previous vehicle states. Furthermore, Kalman filter provides a distance parameter to tolerate distortion and noise of object movements. In short, Kalman filter allows the system to track multiple vehicles while maintaining some prediction and tolerance to the

[1] http://www.mathworks.com/help/images/ref/bwareaopen.html

Figure 6.8: Vehicle tracking with Kalman filter.

complex vehicle motions, such as lane switching and variable acceleration. Figure 6.8 displays a sample screenshot from the vehicle tracking software which assigns a unique ID to each moving vehicle based on past data (only the fast moving lane is being tracked in this example). The next step is to estimate the real-world distance and then calculate the speed of the vehicles based on the tracking information.

6.4.1 Proposed Vehicle Speed Calculation

The most commonly adopted solution for vehicle speed detection is using LIDAR or RADAR devices along with surveillance cameras. One significant drawback of such systems is their lack of ability to determine the correct overspeeding vehicle in some cases, because of the communication gap between the sensor and the camera used for saving the vehicle image. For instance, the camera could capture an image with more than one vehicle in the scene (including the overspeeding vehicle) upon activation by the RADAR signal. Another problem is that the LIDAR/RADAR device accuracy is highly affected by interference from large vehicles. The proposed composite vision system avoids the above problems by adopting a vision-based solution that is able to gather speed information of

a vehicle along with its corresponding image.

Since the vehicles appear to move faster from far to near field due to the perspective projection, direct speed calculation from pixel coordinates is impractical. Therefore, we propose an algorithm to transform the composite image location to the real-world location and determine the vehicle speed in kilometer per hour. The speed calculation involves four main steps:

1. Transform stitched log-polar space coordinates to single individual log-polar space coordinates.

2. Transform individual log-polar space coordinates to camera Cartesian space coordinates.

3. Transform camera Cartesian space coordinates to real world coordinates.

4. Use tracking time information and real world location to calculate vehicle speed.

Step 1: An arbitrary position (u, v) in the stitched log-polar coordinates can be transformed to the individual log-polar space coordinate (U,V) by the following relation, where $u_{InnerRingCrop}$ is the number of rings cropped out, $v_{LowerWedgeCrop}$ is the number of wedges cropped out, and $u_{stitchline}$ is the position of the stitch line.

1. When (u, v) falls in the far-field view range, that is, on the left of the stitching line

$$U = u + u_{InnerRingCrop} \tag{6.3}$$

$$V = v + v_{LowerWedgeCrop} \tag{6.4}$$

2. When (u, v) falls in the near-field view range, that is, on the right of the stitching line

$$U = u + u_{stitchline} + u_{InnerRingCrop} \tag{6.5}$$

$$V = v + v_{LowerWedgeCrop} \tag{6.6}$$

Step 2: The method to transform the individual log-polar space coordinates (U, V) to their corresponding camera Cartesian space coordinate (x, y) is defined as follows:

$$Distance = r_{min} \times e^{\frac{U \times log(\frac{r_{max}}{r_{min}})}{n_r - 1}} \tag{6.7}$$

$$Angle = V \times \frac{2\pi}{n_w} \tag{6.8}$$

$$x = Distance \times \cos(Angle) + x_c \tag{6.9}$$

$$y = Distance \times \sin(Angle) + y_c \tag{6.10}$$

Where n_r represents the number of rings, n_w is number of wedges, x_c and y_c are chosen road vanishing point position, r_{max} and r_{min} are maximum and minimum radii used in the stitching process. Figure 6.9 shows a few instances of this transformation between the two coordinates.

Step 3: The transformation from camera Cartesian space coordinates to real-world coordinates (x, z) follows the method proposed by Wu [113]. Taking into consideration the composite cameras height above the road and tilt angle θ from the road's forward direction, the real-world locations x (transverse direction on the road surface) and z (longitudinal or forward direction on the road surface) can be obtained. To validate the transformation result, the lane markers separation

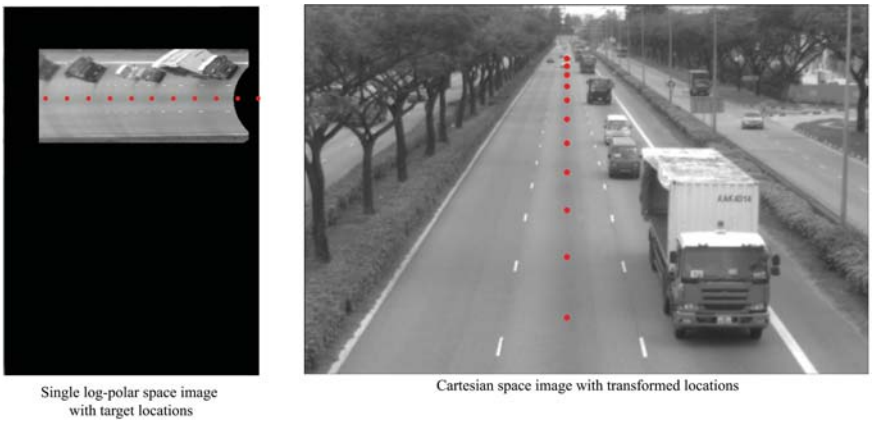
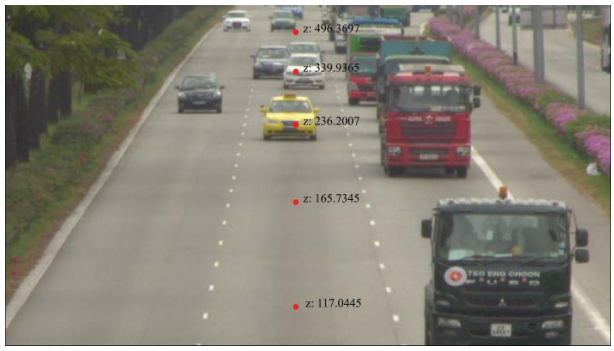


Figure 6.9: An example of the second step of the speed calculation algorithm.

distance (12 m) defined by international traffic standards is used. As shown in Figure 6.10(a), the estimated distances roughly matches with the real-world distances. The accuracy of the distance calculation is further verified using Google maps as shown in Figure 6.10(b). This is the location displayed in Fig. 6.2, which shows a calculated distance of 918.5m from the overhead bridge to the furthest recognizable feature of the road in the far-field camera. Note that the distance information of the LPT space can be calculated using the first stitched frame alone. Hence, every pixel in the LPT space corresponds to a unique real world coordinate. This information is available to the tracking system after the first frame of the video is processed, and hence the calculation of speed can be done more efficiently for the later frames.

Step 4: Using the above transformation, the distance traveled by a vehicle in the real world can be calculated based on the Euclidean distance between any two points of interest. Subsequently, the speed calculation is achieved using the timing information from the vehicle tracking process and the calculated distance. Twenty calculation windows were used to average out the calculated speed of a vehicle. Figure 6.11 shows a sample screenshot of the speed calculation software,



(a) Verification using separation of the lane markers.

(b) Verification using Google Maps.

Figure 6.10: Verification of the real-world distance calculation.

Figure 6.11: An example of the speed calculation step (km/hr).

which includes an instance of the speed calculation being stalled for the first few frames of tracking to ensure reliable estimation. In the next section, we present details about the implementation of the license plate detection system.

6.5 License Plate Detection

After implementing the composite camera, the next step is to employ another camera for the purpose of license plate capturing. Similar to the composite

camera design, the third camera also uses USB 3.0 standards. It uses a lens with a narrow view angle that can focus up to a distance of 75 meters from the shooting point for clear license plate capturing. Furthermore, the camera offers high resolution grayscale images, which suits this application. Figure 6.5(b) shows the complete hardware setup of the traffic monitoring system proposed in this dissertation.

The tracking information is used to trigger the third camera that captures the license plate information whenever a vehicle is detected to be exceeding the allowed speed threshold. In particular, the over-speeding vehicle's position is detected within the visibility range of the license plate capturing camera, and then it is triggered to capture a video. Using the captured video, the overspeeding vehicle is localized using the algorithm proposed by Rosten and Drummond [213], which makes use of point-based and edge-based tracking systems to robustly track fast moving objects. Moreover, the vehicle detection algorithm of [213] performs full-frame feature detection at 400Hz and uses on-line learning for improved performance of feature tracking, which is very suitable for the application proposed in this thesis. Figure 6.12 shows the output of the tracker with the license plate capturing camera adjusted to focus on the inner lane for detecting fast moving vehicles. Notice that the original Cartesian space video is directly used by the license plate detection system, and therefore a more sophisticated tracker has been used which, however, can handle only a single object at a time.

After localizing the overspeeding vehicle, we extract the number plate information by making use of common techniques [214, 215] in computer vision for reading the number plate information. These techniques analyze horizon-

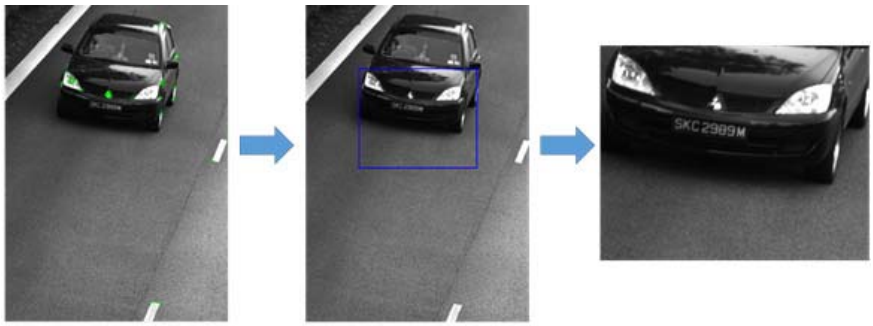


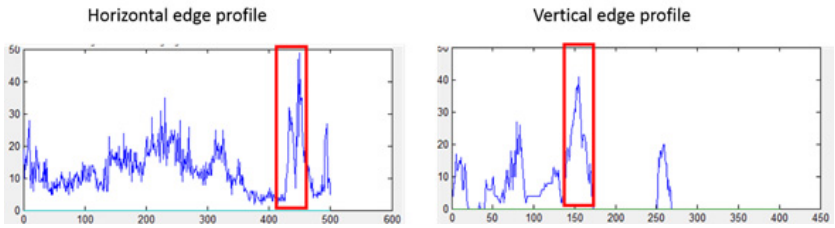
Figure 6.12: An example of the vehicle detection system for extracting the license plate from the Cartesian video. The corner points (marked in green in the leftmost image) are used to spot the most probable area of the moving vehicle in each frame of the video. The image enclosed by the bounding box is subsequently used for extracting the license plate.

tal and vertical edges of an input image to locate the license plate. First, the grayscale image is inverted to obtain an edge image using the Roberts cross operator (see Fig. 6.13(a)). Since a license plate is usually a rectangular region with alphanumeric characters in a plain background, the histogram value for the horizontal and vertical edges of the license plate region will be high, as shown in Fig. 6.13(b). The highlighted peak reveals the probable license plate location and the segmented license plate image is shown in Fig. 6.13(c), which is the final output of the traffic monitoring system proposed in this thesis. In the next section, we present the experimental setup and discuss the results of our system.

6.6 Experimental Setup and Results

The proposed traffic monitoring system has a tracking range of up to one kilometer in practice. Hence, a straight highway with less obstruction by other overhead bridges is desirable. One such location is the overhead bridge after Yuan Ching Road with a maximum visible distance of about 920 meters (verified using Google Maps (Figure 6.10(b)) as well as the distance calculation method

(a) Image inversion and edge detection.



(b) Horizontal and vertical histograms.

(c) Extracted license plate.

Figure 6.13: License plate detection using computer vision techniques.

proposed in section 6.4.1). Figure 6.14(b) shows the field test conducted on an overhead bridge 4.5 m above the expressway lanes. Note that the third camera for license plate detection is placed further to the left of the visible picture to focus on the inner lane. The main challenge for a vision-based traffic monitoring system is to match the short-range reliability of RADAR based systems, and thus it is very important to verify the estimated speed using such systems.

Since the distance calculation has been verified conclusively, the bottleneck is the performance of the Kalman filter. For instance, we observed that if there are nearby trees, which cast shifting shadows on the road, they can be mistaken for a moving vehicle and can also affect the calculated speed when the correctly tracked vehicle enters the shadow of the trees. The wrongly estimated speed (usually higher than the actual speed) is due to the sudden change in the tracked position of the vehicle. Moreover, tracking multiple objects poses sev-

Figure 6.14: Composite Camera Field Test.

eral additional challenges: (1) Multiple detections should be associated with the correct vehicle IDs, (2) New vehicles appearing in the scene should be assigned a suitable ID without confusing with the data of current and past vehicle IDs, (3) Object identity must be maintained when adjacent vehicles merge into a single detection, and also when there is a partial detection of a vehicle due to change in resolution between the near and far field camera. Therefore, extensive experimentation was required in order to come up with a set of suitable configuration parameters.

Furthermore, the motion model used for Kalman filter should ideally correspond to the physical characteristics of the vehicle motion. In reality, most vehicles can be observed to have a complex acceleration profile rather than a constant velocity profile due to traffic conditions. Therefore, a constant acceleration model is a better choice. If the constant velocity model[2] is adopted, the vehicle's location will be quite different from the predicted location, and the tracking results would be sub-optimal no matter what values are selected for the

[2]refer to the demo at http://www.mathworks.com/help/vision/examples/using-kalman-filter-for-object-tracking.html

other parameters. However, even the constant acceleration model does not reflect the true behavior of the expressway vehicles, and therefore the motion noise in terms of location, velocity, and acceleration need to be taken into account. After extensive experiments, we set the maximum location variance, velocity variance and the acceleration variance to be 25, 20, and 10 units respectively. Once suitable configuration parameters are set for the Kalman filter, the next step is to verify the speed using RADAR devices and smartphone applications.

6.6.1 On-site experimental results

Initially, to test the compatibility of the various components of the proposed traffic monitoring system, we arranged a car to drive with a known speed around 75 km/h and then recorded the video using our composite camera setup. With the recorded video, the composite vision system tracked the speed information with 20 calculation windows and calculated the average vehicle speed to be 78 km/h. This is a reasonable speed detection result, because the car's original speed was slightly varied (±5 km/h) due to driving conditions on the expressway. Next, we made use of a calibrated android application to quantitatively verify the estimated speed.

Efforts were made to loan a speed gun or a similar device from the Land Transport Authority of Singapore (LTA) and the Singapore Traffic Police, but both authorities informed that the usage of such device by the public, even for research purposes, is not allowed. Hence, we made use of publicly installed RADAR speed notification boards (Fig. 6.15(a)) to calibrate a smartphone app, and then used it for verifying the estimated speed of the proposed system. The speed board provides reference speed data for comparing the output of mobile

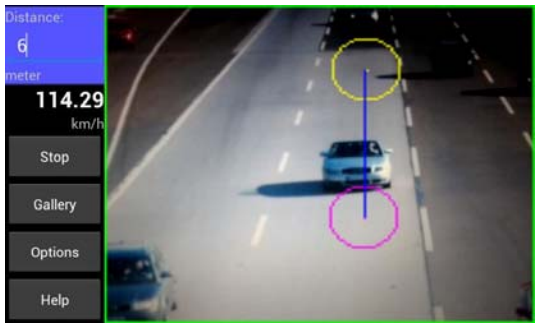
applications, several of which were tested for suitability and accuracy. Finally, an app called as Speed Radar Cam was found to be the most accurate and reliable mobile application to aid in verifying the speed calculation algorithm. A screenshot of the Speed Radar Cam is shown in Fig. 6.15(b).

The mobile application was used during the on-site experiments to verify the accuracy of the tracking software. The errors between the calculated speed and the output from Speed Radar Cam usually deviate between 3km/h. Hence, the accuracy of the proposed speed calculation algorithm falls within a reasonable range and further accuracy verification would require devices such as a speed gun or a portable RADAR/LIDAR system. A sample output of the Speed Radar Cam is shown in Fig. 6.15(c).

After establishing the reliability of the speed estimation, we tested the proposed traffic monitoring system on a variety of traffic conditions. Figure 6.16 shows the results from a video captured at Ayer Rajah Expressway in Singapore; the upper half of the MATLAB GUI displays the tracking result while the lower half shows the calculated speed. After each vehicle exits the scene, a snapshot of it is stored along with the time stamp and average speed. Notice that NaN appears as one of the vehicles' speed in Fig.6.16. This is because the speed calculation algorithm waits for tracking to stabilize, which typically takes a couple of seconds.

We conducted a total of 9 video recordings, each of which contains about 15 to 30 vehicles in the fast lane. It was observed that the speed limit of 90km/hr on the Ayer Rajah expressway was not violated by 95% of the vehicles tracked by the composite vision system, and most of the cars that violated the speed limit

(a) The RADAR speed detection system used for calibrating the smartphone application.



(b) Screenshot of the Speed Radar Cam application.

(c) Sample output of the Speed Radar Cam application. Top left shows the speed of the nearest vehicle.

Figure 6.15: Verification of the speed calculation.

were estimated to be around 100 km/hr while in reality they were traveling at a touch above or below the speed limit. The main cause for the higher estimation of speed was due to the case of adjacent vehicles merging into a single detection and when there is a partial detection, both of which change the centroid location of the vehicle in LPT space by a considerable margin. Another concern was the shadows cast on the road by big trees, which typically is not stationary in windy conditions, and some unreasonable speed values were obtained (above 150 km/hr). Of course, these anomalous speed values can be filtered out and

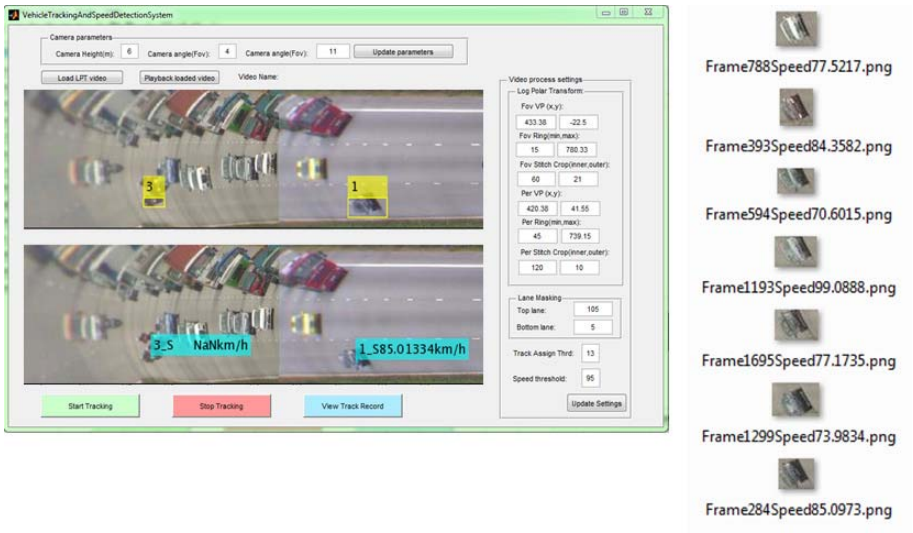


Figure 6.16: Results of the composite vision system.

the speed values calculated in that region were not taken into account while calculating the average speed. Next, we report the results of the license plate detection system. Figure 6.17 shows a sample output of the complete traffic monitoring system. The success of the license plate detection depends on the uniqueness of the edge profile of the number plate. In some cases, the edges of the number plate do not stand out from their neighborhood and this results in no segmentation. However, since there are many frames of the vehicle, there is usually at least one output with a clear license plate segmentation. In the extreme case where there is no segmentation, the system simply records the speed along with the image obtained by the tracker (Fig. 6.12). It is possible to opt for more sophisticated license plate detectors, like specifically trained object detectors [216], but we kept the algorithm simple during the nascent stages of this work.

Figure 6.17: Results of the composite vision system integrated with the license plate detection module.

6.6.2 Discussion

Since the vehicle motion is tracked from far field to near field, the composite camera provides adequate information for the system to perform long distance tracking and speed calculation. This benefits the accuracy and reliability of the speed calculation. With the multiple depth-of-field viewing ability, the system can track vehicles up to 1000 m away from the shooting point, which is a big improvement compared to conventional practices up to 300 m using vision-based methods. Various parameters need to be considered for the speed calculation procedure, such as road lane vanishing point setting, log-polar transform parameters, and individual camera settings. However, the speed calculation is most affected by the precision of object tracking. From experiments in other parts of the island, we found that another challenging aspect was low lighting condi-

tions. In these cases, the speed detection result was not reliable due to tracking difficulties.

As pointed out earlier, the main drawback of the RADAR based traffic monitoring systems is the lack of vehicle identity information while estimating the speed. The composite vision system solves this problem by providing a snapshot of the exact overspeeding vehicle, as shown in Figure 6.16.

6.7 Summary

We proposed a composite vision system with multiple depth-of-field viewing ability that largely extended the tracking range of traditional traffic monitoring systems. By defining the overspeeding vehicle using the tracking result, strong coherence between identity and speed information was established. The addition of a separate license plate detection camera to the composite vision system provides sufficient evidence for law enforcement. Having deep field object tracking ability, the composite vision system can handle high-speed vehicle tracking and can compensate the drawbacks of existing speed monitoring systems. Moreover, the system has the potential to perform real-time tracking in complex road conditions and multiple lanes. It's simultaneous near and far sensing capabilities can also be extended to other industries, such as faulty item inspection along a conveyor belt in manufacturing industries as well as employment in Unmanned Aerial Vehicles (UAV).

Chapter 7

Conclusions

Ever since our computers have achieved a crude understanding of images, computer vision has profoundly changed our lives in many ways. Applications such as image database search in the internet, computational photography, biological imaging, vision for graphics, geographical information system, biometrics, vision for nanotechnology, etc., were unanticipated while other applications keep arising as computer vision technology diversifies. Rapid developments in supportive technologies, such as digital cameras and computers, ensure that computer vision systems will become increasingly more capable and affordable. Among the various topics in this exciting field of research, we mainly focused on the important problem of classifying object images in this dissertation.

Firstly, the problem of classifying shapes of objects, even with proper segmentation, is very challenging in the face of occlusion and strong view-point changes. While many valuable results have been obtained considering the global shape image, there has been little effort to consider local shape features with successful classification schemes such as the bag-of-words model. In this the-

sis, we proposed a novel local shape descriptor using log-polar transform to deal with scale, rotation and view-point variations. Using the proposed features along with contextual information, we demonstrated much better classification performance compared to state-of-the-art shape classification algorithms on the animal shapes dataset. Secondly, we considered the more general problem of classifying grayscale images. While appearance based features have drawn most of the attention in the past two decades, a few works have considered integrating shape and appearance cues. In this dissertation, we proposed a novel fusion of appearance and shape cues using log-polar transform, and demonstrated significantly higher performance compared to existing works on the ETH-80 dataset. Thirdly, we showed that high performance can be achieved by integrating color, appearance and shape cues on two popular object datasets. Finally, we proposed a real-world application of log-polar transform for tracking high-speed moving objects.

7.1 Main Contributions

In Chapter 3, we investigated the classification of binary shape images with scale, rotation and strong view-point variations, based on features derived using log-polar transform. Different from most of the existing works, we considered a local feature based classification using the bag-of-words model, which has been rarely applied to shape classification. It was found that, with contextual information encoded in the image representation, the performance of the shape classification system was significantly better than the state-of-the-art algorithms on the animal shapes dataset. Besides the above contributions, a novel metric

termed 'weighted gain ratio' was proposed to select a suitable codebook size in the bag-of-words model. The proposed metric is generic, and hence it can be used for any clustering quality evaluation task. Additionally, a joint learning framework was proposed to learn features in a data-driven manner, and thus avoiding manual fine-tuning of the model parameters.

In Chapter 4, we investigated the classification of grayscale images based on log-polar encoded local features extracted from different object cues. To extract different object cues, we proposed a novel scheme to obtain structure, texture and shape information from grayscale images. The extracted local descriptors were quantized using the bag-of-words representation with two key contributions. First, a keypoint detection scheme based on image denoising was proposed to select sampling locations, which was shown to outperform the widely used dense grid sampling by a large margin. Second, a codebook optimization scheme based on discrete entropy was proposed to reduce the number of codewords and at the same time increase the overall performance. The proposed cue-based object categorization framework was demonstrated to have significantly higher classification performance compared to existing works on the widely used ETH-80 object dataset.

To extend the classification framework to color images, we proposed a novel multi-cue object representation using the bag-of-words model in Chapter 5. Majority of the existing works focus on advanced encoding methods or sophisticated feature pooling techniques or machine learning strategies to obtain better performance over the simple bag-of-words model. In contrast, we proposed log-polar encoded local features while still employing the original bag-of-words represen-

tation (vector quantization). Besides the proposed features, we introduced a novel keypoint detection method that was found to be better than the dense sampling strategy from a practical point of view. In other words, we demonstrated par performance compared to the dense sampling strategy at a much lower computational cost. Thus, the proposed keypoint detection scheme using differential entropy, offers a more principled approach to image sampling for the popular bag-of-words framework. Finally, we proposed a novel way to encode contextual information in the bag-of-words model, which improves the overall accuracy without affecting the dimensionality of the features in a significant way. The proposed multi-cue object representation was shown to outperform seminal works on the popular Caltech-101 object dataset. In addition, we outperformed several state-of-the-art methods on the Flickr-101 object dataset.

In Chapter 6, we designed a video processing application based on the log-polar sampling technique extensively used in the earlier chapters. In particular, log-polar transform was used to stitch video information acquired from cameras of different visual field depths into a single video stream. Consequently, it was possible to track moving objects with a much longer tracking range (3 times longer) compared to using a single camera. This composite vision system was applied to the problem of traffic monitoring in expressways. Having a deep field object tracking ability, the composite vision system was able to handle high-speed vehicle tracking, and thus compensate the drawbacks of current speed monitoring systems. Moreover, the addition of a separate camera for license plate detection provided sufficient evidence for law enforcement. The experimental results demonstrated the effectiveness of the proposed traffic monitoring system.

7.2 Suggestions for Future Work

Based on the research presented in this dissertation, the following issues deserve further consideration and investigation.

1. The proposed cue-based object classification framework can be extended to incorporate other visual cues. While we made some progress to include multiple object cues in Chapters 4 and 5, it is important to include other appearance cues like depth, which can also assist in improving the shape cue. As a starting point, it is a possible direction to use monocular depth extraction techniques like the one proposed in [217]. Additionally, extending our classification framework to large-scale and multi-label object classification problems like the PASCAL VOC Challenge is a potential direction.

2. The performance of the shape classification system was found to be suboptimal compared to the performance of other object cues, as seen in Chapter 5. There are many ways to extract shape information from grayscale and color images. For instance, the choice of the salient object detection algorithm or the thresholding method profoundly affects the quality of the extracted shape. Since this is the most important problem in computer vision, i.e., object segmentation, it is desirable to combine mutually informative tasks, such as depth estimation and shape extraction, to be more effective.

3. Even though incorporating contextual information improved the classification accuracy, as shown in Chapters 3 and 5, it was only used for the shape

cue. As discussed in Chapter 5, incorporation of contextual information for the appearance cues is extremely prohibitive in terms of memory requirements without compromising on classification accuracy. Therefore, for dealing with large codebooks, data mining methods can be used to choose particular codewords to build a smaller co-occurrence matrix. Moreover, it is a potential direction to study the performance of contextual information for appearance cues using methods other than co-occurrence statistics.

4. For the vision-based traffic monitoring application proposed in Chapter 6, a possible future work is to attempt a hybridization with RADAR systems, which will help complement their respective strengths in traffic monitoring. Also, an optical character recognition (OCR) program can be included to automatically read the license plate image and output the corresponding characters to fully automate the law enforcement procedure. Moreover, a fast machine code implementation that can handle real-time object tracking in multiple lanes would be a potential direction of research.

In conclusion, developing accurate and fast object classification systems is very important since they are an integral aspect of many practical computer vision systems. To achieve the objective of making machines perceive the world as humans do, we believe it is important to take measured steps to integrate visual cues in a unified classification framework. This dissertation represents a step in this direction.

Bibliography

[1] L Roberts. Pattern recognition with an adaptive network. In *Proc. IRE International Convention Record*, pages 66–70, 1960.

[2] Elias N Malamas, Euripides G.M Petrakis, Michalis Zervakis, Laurent Petit, and Jean-Didier Legat. A survey on industrial vision systems, applications and tools. *Image and Vision Computing*, 21(2):171 – 188, 2003.

[3] Shunji Mori, Hirobumi Nishida, and Hiromitsu Yamada. *Optical character recognition*. John Wiley & Sons, Inc., 1999.

[4] Masakazu Ejiri. Machine vision in early days: Japan's pioneering contributions. In *Asian Conference on Computer Vision*, volume 4843 of *Lecture Notes in Computer Science*, pages 35–53, 2007.

[5] Michael I Posner, Mary J Nissen, and Raymond M Klein. Visual dominance: an information-processing account of its origins and significance. *Psychological review*, 83(2):157, 1976.

[6] J.L. Crowley and Alice C. Parker. A representation for shape based on peaks and ridges in the difference of low-pass transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(2):156–170, 1984.

[7] J.L. Crowley and Arthur C. Sanderson. Multiple resolution representation and probabilistic matching of 2-D gray-scale shape. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(1):113–121, 1987.

[8] I. Biederman. Recognition-by-components: a theory of human image understanding. *Psychological review*, 94(2):115, 1987.

[9] Tinne Tuytelaars and Krystian Mikolajczyk. Local invariant feature detectors: a survey. *Found. Trends. Comput. Graph. Vis.*, 3(3):177–280, 2008.

[10] Alexander Andreopoulos and John K. Tsotsos. 50 years of object recognition: Directions forward. *Computer Vision and Image Understanding*, 117(8):827 – 891, 2013.

[11] D.G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 2, pages 1150–1157, 1999.

[12] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 886 –893, 2005.

[13] Farzin Mokhtarian and Alan Mackworth. Scale-based description and recognition of planar curves and two-dimensional shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(1):34 –43, 1986.

[14] William J. Rucklidge. Efficiently locating objects using the hausdorff distance. *International Journal of Computer Vision*, 24:251–270, 1997.

[15] Dengsheng Zhang and Guojun Lu. A comparative study of fourier descriptors for shape representation and retrieval. In *Proc. of 5th Asian Conference on Computer Vision (ACCV)*, pages 646–651, 2002.

[16] Herbert Freeman. On the encoding of arbitrary geometric configurations. *IRE Transactions on Electronic Computers*, 10(2):260 –268, 1961.

[17] R. Chellappa and R. Bagdazian. Fourier coding of image boundaries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(1):102 –105, 1984.

[18] Jukka Livarinen and Ari Visa. Shape recognition of irregular objects. In *Intelligent Robots and Computer Vision XV: Algorithms, Techniques, Active Vision, and Materials Handling,*, pages 25–32, 1996.

[19] A. Del Bimbo and P. Pala. Visual image retrieval by elastic matching of user sketches. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(2):121 –132, 1997.

[20] Haruo Asada and Michael Brady. The curvature primal sketch. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(1):2 –14, 1986.

[21] Gregory Dudek and John K. Tsotsos. Shape representation and recognition from multiscale curvature. *Comput. Vis. Image Underst.*, 68(2):170–189, 1997.

[22] Ming-Kuei Hu. Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory*, 8(2):179 –187, 1962.

[23] Ardeshir Goshtasby. Description and discrimination of planar shapes using shape matrices. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7(6):738 –743, 1985.

[24] Richard J Prokop and Anthony P Reeves. A survey of moment-based techniques for unoccluded object representation and recognition. *CVGIP: Graphical Models and Image Processing*, 54(5):438 – 460, 1992.

[25] Whoi-Yul Kim and Yong-Sung Kim. A region-based shape descriptor using Zernike moments. *Signal Processing: Image Communication*, 16:95 – 102, 2000.

[26] David McG. Squire and Terry M. Caelli. Invariance signatures: Characterizing contours by their departures from invariance. *Computer Vision and Image Understanding*, 77(3):284 – 316, 2000.

[27] Dengsheng Zhang and Guojun Lu. Review of shape representation and description techniques. *Pattern Recognition*, 37(1):1 – 19, 2004.

[28] Byung-Woo Hong and S. Soatto. Shape matching using multiscale integral invariants. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(1):151–160, 2015.

[29] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509 –522, 2002.

[30] E. Shechtman and M. Irani. Matching local self-similarities across images and videos. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1 –8, 2007.

[31] U. Schmidt and S. Roth. Learning rotation-aware features: From invariant priors to equivariant descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2050–2057, 2012.

[32] Richard A Messner and Harold H Szu. An image processing architecture for real time generation of scale and rotation invariant patterns. *Computer vision, graphics, and image processing*, 31(1):50–66, 1985.

[33] S. Zokai and G. Wolberg. Image registration using log-polar mappings for recovery of large-scale similarity and projective transformations. *IEEE Transactions on Image Processing*, 14(10):1422–1434, 2005.

[34] R. Matungka, Y.F. Zheng, and R.L. Ewing. Object recognition using log-polar wavelet mapping. In *IEEE International Conference on Tools with Artificial Intelligence*, pages 559 –563, 2008.

[35] D. Young. Straight lines and circles in the log-polar image. In *British Machine Vision Conf.*, pages 426–435, 2000.

[36] I. Kokkinos and A. Yuille. Scale invariance without scale selection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1 –8, 2008.

[37] Joseph G Bailey and Richard A Messner. Log-polar mapping as a preprocessing stage for an image tracking system. In *Robotics Conferences*, pages 15–22, 1989.

[38] Fredrik Viksten and Anders Moe. Local single-patch features for pose estimation using the log-polar transform. In *Iberian conference on Pattern Recognition and Image Analysis*, pages 44–51, 2005.

[39] Gabriella Csurka, Christopher R. Dance, Lixin Fan, Jutta Willamowski, and Cdric Bray. Visual categorization with bags of keypoints. In *In Workshop on Statistical Learning in Computer Vision, ECCV*, pages 1–22, 2004.

[40] S. Lazebnik, C. Schmid, and J. Ponce. A sparse texture representation using local affine regions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1265 –1278, 2005.

[41] P. Quelhas, F. Monay, J.-M. Odobez, D. Gatica-Perez, T. Tuytelaars, and L. Van Gool. Modeling scenes with local descriptors and latent aspects. In *Tenth IEEE International Conference on Computer Vision*, pages 883 – 890, 2005.

[42] Anna Bosch, Andrew Zisserman, and Xavier Munoz. Scene classification via pLSA. In *European Conference on Computer Vision*, volume 3954 of *Lecture Notes in Computer Science*, pages 517–530, 2006.

[43] S. Agarwal, A. Awan, and D. Roth. Learning to detect objects in images via a sparse, part-based representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(11):1475 –1490, 2004.

[44] Thomas Leung and Jitendra Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *International Journal of Computer Vision*, 43:29–44, 2001.

[45] Ankur Agarwal and Bill Triggs. Hyperfeatures - multilevel local coding for visual recognition. In *European Conference on Computer Vision*, pages 30–43, 2006.

[46] K. Grauman and T. Darrell. Efficient image matching with distributions of local invariant features. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 627 – 634, 2005.

[47] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 264 – 271, 2003.

[48] J. Winn, A. Criminisi, and T. Minka. Object categorization by learned universal visual dictionary. In *Tenth IEEE International Conference on Computer Vision*, volume 2, pages 1800 –1807, 2005.

[49] Kart-Leong Lim and H.K. Galoogahi. Shape classification using local and global features. In *Fourth Pacific-Rim Symposium on Image and Video Technology (PSIVT)*, pages 115 –120, 2010.

[50] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2169–2178, 2006.

[51] Noha M. Elfiky, Jordi Gonzalez, and F. Xavier Roca. Compact and adaptive spatial pyramids for scene recognition. *Image and Vision Computing*, 30(8):492 – 500, 2012.

[52] Jianchao Yang, Kai Yu, Yihong Gong, and Thomas Huang. Linear spatial pyramid matching using sparse coding for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 1794–1801, 2009.

[53] Y. Jia, C. Huang, and T. Darrell. Beyond spatial pyramids: Receptive field learning for pooled image features. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3370–3377, 2012.

[54] Qiang Chen, Zheng Song, Yang Hua, Zhongyang Huang, and Shuicheng Yan. Hierarchical matching with side information for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3426–3433, 2012.

[55] Eric Nowak, Frdric Jurie, and Bill Triggs. Sampling strategies for bag-of-features image classification. In *European Conference on Computer Vision*, volume 3954 of *Lecture Notes in Computer Science*, pages 490–503, 2006.

[56] Carolina Galleguillos and Serge Belongie. Context based object categorization: A critical survey. *Comput. Vis. Image Underst.*, 114(6):712–722, 2010.

[57] Jianguo Li, Weixin Wu, Tao Wang, and Yimin Zhang. One step beyond histograms: Image representation using Markov stationary features. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1 –8, 2008.

[58] R.M. Haralick, K. Shanmugam, and Its'Hak Dinstein. Textural features for image classification. *IEEE Transactions on Systems, Man and Cybernetics*, 3(6):610–621, 1973.

[59] Christopher D. Manning and Hinrich Schütze. *Foundations of statistical natural language processing*. MIT Press, Cambridge, MA, USA, 1999.

[60] Naeem A Bhatti and Allan Hanbury. Co-occurrence bag of words for object recognition. In *Proceedings of the 15th Computer Vision Winter Workshop*, pages 21–28, 2010.

[61] Tinne Tuytelaars, ChristophH. Lampert, MatthewB. Blaschko, and Wray Buntine. Unsupervised object discovery: A comparison. *International Journal of Computer Vision*, 88:284–302, 2010.

[62] Mingyuan Jiu, Christian Wolf, Christophe Garcia, and Atilla Baskurt. Supervised learning and codebook optimization for bag-of-words models. *Cognitive Computation*, 4:409–419, 2012.

[63] Jingen Liu and Mubarak Shah. Learning human actions via information maximization. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1 –8, 2008.

[64] Jingen Liu, Yang Yang, and M. Shah. Learning semantic visual vocabularies using diffusion distance. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 461 –468, 2009.

[65] J. R. Quinlan. Induction of decision trees. *Mach. Learn.*, 1(1):81–106, 1986.

[66] Fahad S Khan, Joost Weijer, Andrew D Bagdanov, and Maria Vanrell. Portmanteau vocabularies for multi-cue image representation. In *Advances in neural information processing systems*, pages 1323–1331, 2011.

[67] B Leibe, K Mikolajczyk, and B Schiele. Segmentation based multi-cue integration for object detection. In *British Machine Vision Conference*, pages 1169 – 1178, 2006.

[68] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman. Multiple kernels for object detection. In *IEEE International Conference on Computer Vision*, pages 606–613, 2009.

[69] P. Gehler and S. Nowozin. On feature combination for multiclass object classification. In *12th International Conference on Computer Vision*, pages 221–228, 2009.

[70] Fuxiang Lu, Xiaokang Yang, Weiyao Lin, Rui Zhang, and Songyu Yu. Image classification with multiple feature channels. *Optical Engineering*, 50(5):57210–9, 2011.

[71] Jamie Shotton, Andrew Blake, and Roberto Cipolla. Efficiently combining contour and texture cues for object recognition. In *British Machine Vision Conference*, pages 1–10, 2008.

[72] Jamie Shotton, John Winn, Carsten Rother, and Antonio Criminisi. Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context. *Int. Journal of Computer Vision (IJCV)*, 81(1):2–23, 2009.

[73] J. Shotton, A. Blake, and R. Cipolla. Multiscale categorical object recognition using contour fragments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(7):1270–1281, 2008.

[74] Pawan M. Kumar, Philip Torr, and Andrew Zisserman. Extending pictorial structures for object recognition. In *British Machine Vision Conference*, pages 789–798, 2004.

[75] Anna Bosch, Andrew Zisserman, and Xavier Munoz. Representing shape with a spatial pyramid kernel. In *Proceedings of the 6th ACM International Conference on Image and Video Retrieval*, pages 401–408, 2007.

[76] Xinggang Wang, Bin Feng, Xiang Bai, Wenyu Liu, and Longin Jan Latecki. Bag of contour fragments for robust shape classification. *Pattern Recognition*, 47(6):2116 – 2125, 2014.

[77] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5):898–916, May 2011.

[78] Chuan Yang, Lihe Zhang, and Huchuan Lu. Graph-regularized saliency detection with convex-hull-based center prior. *IEEE Signal Processing Letters*, 20(7):637–640, 2013.

[79] A. Buades, B. Coll, and J. Morel. A review of image denoising algorithms, with a new one. *Multiscale Modeling & Simulation*, 4(2):490–530, 2005.

[80] Deqing Sun, S. Roth, and M.J. Black. Secrets of optical flow estimation and their principles. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2432–2439, 2010.

[81] Andreas Wedel, Thomas Pock, Christopher Zach, Horst Bischof, and Daniel Cremers. An improved algorithm for tv-l1 optical flow. In *Statistical and Geometrical Approaches to Visual Motion Analysis*, volume 5604 of *Lecture Notes in Computer Science*, pages 23–45, 2009.

[82] Leonid I. Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60:259 – 268, 1992.

[83] Arturo Ribes, Senshan Ji, Arnau Ramisa, and Ramon López de Mántaras. Self-supervised clustering for codebook construction: An application to object localization. In *Proceedings of the 14th International Conference of the Catalan Association for Artificial Intelligence*, pages 208–217, 2011.

[84] Bharath Ramesh, Cheng Xiang, and Tong Heng Lee. Shape classification using invariant features and contextual information in the bag-of-words model. *Pattern Recognition*, 48(3):894 – 906, 2015.

[85] Sungho Kim and In So Kweon. Object categorization robust to surface markings using entropy-guided codebook. In *IEEE Workshop on Applications of Computer Vision*, pages 22–22, 2007.

[86] Lei Wang, Luping Zhou, and Chunhua Shen. A fast algorithm for creating a compact and discriminative visual codebook. In *European Conference on Computer Vision*, volume 5305 of *Lecture Notes in Computer Science*, pages 719–732, 2008.

[87] Sungho Kim and In So Kweon. Simultaneous classification and visualword selection using entropy-based minimum description length. In *International Conference on Pattern Recognition*, volume 1, pages 650–653, 2006.

[88] MS Livingstone and DH Hubel. Psychophysical evidence for separate channels for the perception of form, color, movement, and depth. *The Journal of Neuroscience*, 7(11):3416–3468, 1987.

[89] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254–1259, 1998.

[90] Ali Borji, DickyN. Sihite, and Laurent Itti. Salient object detection: A benchmark. In *European Conference on Computer Vision*, Lecture Notes in Computer Science, pages 414–429, 2012.

[91] Jinjun Wang, Jianchao Yang, Kai Yu, Fengjun Lv, T. Huang, and Yihong Gong. Locality-constrained linear coding for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3360–3367, 2010.

[92] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[93] H. Jegou, M. Douze, C. Schmid, and P. Perez. Aggregating local descriptors into a compact image representation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3304–3311, 2010.

[94] Xinggang Wang, Xiang Bai, Wenyu Liu, and L.J. Latecki. Feature context for image classification and object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 961–968, 2011.

[95] Ken Chatfield, Victor Lempitsky, Andrea Vedaldi, and Andrew Zisserman. The devil is in the details: an evaluation of recent feature encoding methods. In *Proceedings of the British Machine Vision Conference*, pages 76.1–76.12, 2011.

[96] Jiashi Feng, Bingbing Ni, Dong Xu, and Shuicheng Yan. Histogram contextualization. *IEEE Transactions on Image Processing*, 21(2):778 –788, feb. 2012.

[97] Yin Li, Yue Zhou, Junchi Yan, Zhibin Niu, and Jie Yang. Visual saliency based on conditional entropy. In *Asian Conference on Computer Vision*, volume 5994 of *Lecture Notes in Computer Science*, pages 246–257, 2010.

[98] Xiaodi Hou and Liqing Zhang. Dynamic visual attention: Searching for coding length increments. In *Advances in neural information processing systems*, pages 681–688, 2009.

[99] Neil Bruce and John Tsotsos. Attention based on information maximization. *Journal of Vision*, 7(9):950–950, 2007.

[100] Timor Kadir, Andrew Zisserman, and Michael Brady. An affine invariant salient region detector. In *European Conference on Computer Vision*, volume 3021 of *Lecture Notes in Computer Science*, pages 228–241, 2004.

[101] Timor Kadir and Michael Brady. Saliency, scale and image description. *International Journal of Computer Vision*, 45(2):83–105, 2001.

[102] Tilke Judd, Frédo Durand, and Antonio Torralba. A benchmark of computational models of saliency to predict human fixations. Technical Report TR-2012-001, MIT-CSAIL, 2012.

[103] James M Rehg and Takeo Kanade. Visual tracking of high DOF articulated structures: an application to human hand tracking. In *European Conference on Computer Vision*, pages 35–46, 1994.

[104] Robert JK Jacob and Keith S Karn. Eye tracking in human-computer interaction and usability research: Ready to deliver the promises. *Mind*, 2(3):4, 2003.

[105] Gian Luca Foresti. Object recognition and tracking for remote video surveillance. *IEEE Transactions on Circuits and Systems for Video Technology*, 9(7):1045–1062, 1999.

[106] I. Cohen and G. Medioni. Detecting and tracking moving objects for video surveillance. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, page 325, 1999.

[107] Omar Javed and Mubarak Shah. Tracking and object classification for automated surveillance. In *European Conference on Computer Vision*, pages 343–357, 2002.

[108] S. Saripalli, J.F. Montgomery, and G. Sukhatme. Visually guided landing of an unmanned aerial vehicle. *IEEE Transactions on Robotics and Automation*, 19(3):371–380, 2003.

[109] Benjamin Coifman, David Beymer, Philip McLauchlan, and Jitendra Malik. A real-time computer vision system for vehicle tracking and traffic surveillance. *Transportation Research Part C: Emerging Technologies*, 6(4):271–288, 1998.

[110] Shunsuke Kamijo, Yasuyuki Matsushita, Katsushi Ikeuchi, and Masao Sakauchi. Traffic monitoring and accident detection at intersections. *IEEE Transactions on Intelligent Transportation Systems*, 1(2):108–118, 2000.

[111] Jen-Chao Tai, Shung-Tsang Tseng, Ching-Po Lin, and Kai-Tai Song. Real-time image tracking for automatic traffic monitoring and enforcement applications. *Image and Vision Computing*, 22(6):485–501, 2004.

[112] Liang Wang, Weiming Hu, and Tieniu Tan. Recent developments in human motion analysis. *Pattern recognition*, 36(3):585–601, 2003.

[113] Jianping Wu, Zhaobin Liu, Jinxiang Li, Gu Caidong, Maoxin Si, and Fangyong Tan. An algorithm for automatic vehicle speed detection using video camera. In *4th International Conference on Computer Science Education*, pages 193–196, 2009.

[114] Sedat Doan, Mahir Serhan Temiz, and Stk Klr. Real time speed estimation of moving vehicles from side view images from an uncalibrated video camera. *Sensors*, 10(5):4805, 2010.

[115] Xavier Clady, François Collange, Frederic Jurie, and Philippe Martinet. Cars detection and tracking with a vision sensor. In *Proceedings of the Intelligent Vehicles Symposium*, pages 593–598, 2003.

[116] Christoph Roessing, Axel Reker, Michael Gabb, Klaus Dietmayer, and Hendrik Lensch. Intuitive visualization of vehicle distance, velocity and risk potential in rear-view camera applications. In *Proceedings of the Intelligent Vehicles Symposium*, pages 579–585, 2013.

[117] Chieh-Chih Wang, Charles Thorpe, and Arne Suppe. Ladar-based detection and tracking of moving objects from a ground vehicle at high speeds. In *Proceedings of the Intelligent Vehicles Symposium*, pages 416–421, 2003.

[118] Benjamin Kormann, Antje Neve, Gudrun Klinker, Walter Stechele, et al. Stereo vision based vehicle detection. In *International Conference on Computer Vision Theory and Applications (VISAPP)*, volume 2, pages 431–438, 2010.

[119] Gwenaëlle Toulminet, Massimo Bertozzi, Stéphane Mousset, Abdelaziz Bensrhair, and Alberto Broggi. Vehicle detection by means of stereo vision-based obstacles features extraction and monocular pattern analysis. *IEEE Transactions on Image Processing*, 15(8):2364–2375, 2006.

[120] Pavlo B. Melnyk Richard A. Messner. Mobile digital video system for law enforcement. In *Vehicular Technology Conference*, pages 468–472, 2002.

[121] Todd N Schoepflin and Daniel J Dailey. Dynamic camera calibration of roadside traffic management cameras for vehicle speed estimation. *IEEE Transactions on Intelligent Transportation Systems*, 4(2):90–98, 2003.

[122] Ben Simmoneau. I-team: Controversy over speed cameras. CBS News, March 26 2013.

[123] Motor Defence Solicitors. Guide to speed detection devices. Motor Defence Team Webpage, Oct 2014.

[124] Pavlo Melnyk. *Biologically inspired composite image sensor for deep field target tracking.* PhD thesis, University of New Hampshire, 2008.

[125] Jean-Michel Morel and Guoshen Yu. Is SIFT scale invariant? *Inverse Problems and Imaging*, 5(1):115–136, 2011.

[126] Eric L Schwartz. Spatial mapping in the primate sensory projection: analytic structure and relevance to perception. *Biological cybernetics*, 25(4):181–194, 1977.

[127] Huang Dong. *Discriminant feature analysis for pattern recognition.* PhD thesis, National University of Singapore, 2010.

[128] R.O. Duda, P.E. Hart, and D.G. Stork. Pattern classification. *New York: John Wiley*, 4:170, 2001.

[129] Matthew Turk and Alex Pentland. Eigenfaces for recognition. *Journal of cognitive neuroscience*, 3(1):71–86, 1991.

[130] P.N. Belhumeur, J.P. Hespanha, and D. Kriegman. Eigenfaces vs. Fisherfaces: recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997.

[131] C. Xiang, X.A. Fan, and T.H. Lee. Face recognition using recursive Fisher linear discriminant. *IEEE Transactions on Image Processing*, 15(8):2097 –2105, 2006.

[132] N. Thakoor, J. Gao, and Sungyong Jung. Hidden Markov model-based weighted likelihood discriminant for 2-D shape classification. *IEEE Transactions on Image Processing*, 16(11):2707 –2719, 2007.

[133] T.B. Sebastian, P.N. Klein, and B.B. Kimia. Recognition of shapes by editing their shock graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(5):550–571, 2004.

[134] G. Andreu, A. Crespo, and J.M. Valiente. Selecting the toroidal self-organizing feature maps (TSOFM) best organized to object recognition. In *International Conference on Neural Networks*, volume 2, pages 1341–1346, 1997.

[135] N. Thakoor and J. Gao. Shape classifier based on generalized probabilistic descent method with hidden Markov descriptor. In *IEEE International Conference on Computer Vision (ICCV)*, volume 1, pages 495 – 502, 2005.

[136] N. Thakoor, Sungying Jung, and Jean Gao. Hidden Markov model based weighted likelihood discriminant for minimum error shape classification. In *IEEE International Conference on Multimedia and Expo*, pages 342–345, 2005.

[137] Manuele Bicego, Vittorio Murino, and Mrio A.T. Figueiredo. Similarity-based classification of sequences using hidden Markov models. *Pattern Recognition*, 37(12):2281 – 2291, 2004.

[138] Pietro Lovato and Manuele Bicego. 2D shapes classification using BLAST. In *Structural, Syntactic, and Statistical Pattern Recognition*, volume 7626 of *Lecture Notes in Computer Science*, pages 273–281, 2012.

[139] Xiang Bai, Xingwei Yang, Deguang Yu, and Longin Jan Latecki. Skeleton-based shape classification using path similarity. *International Journal of Pattern Recognition and Artificial Intelligence*, 22(04):733–746, 2008.

[140] K.B. Sun and B.J. Super. Classification of contour shapes using class segment sets. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 727–733, 2005.

[141] B. Wang, W. Shen, W. Y Liu, X. G You, and X. Bai. Shape classification using tree -unions. In *20th International Conference on Pattern Recognition (ICPR)*, pages 983–986, 2010.

[142] Vassilis G. Kaburlasos, S.E. Papadakis, and Angelos Amanatiadis. Binary image 2D shape learning and recognition based on lattice-computing (LC) techniques. *Journal of Mathematical Imaging and Vision*, 42(2-3):118–133, 2012.

[143] Mohammad Reza Daliri and Vincent Torre. Shape recognition based on kernel-edit distance. *Computer Vision and Image Understanding*, 114(10):1097 – 1103, 2010.

[144] M. Daliri and Vincent Torre. Robust symbolic representation for shape recognition and retrieval. *Pattern Recognition*, 41(5):1782 – 1798, 2008.

[145] Michel Neuhaus and Horst Bunke. Edit distance-based kernel functions for structural pattern classification. *Pattern Recognition*, 39(10):1852 – 1863, 2006.

[146] Teuvo Kohonen. The self-organizing map. *Neurocomputing*, 21(1):1 – 6, 1998.

[147] Michael S. Lew, Nicu Sebe, Chabane Djeraba, and Ramesh Jain. Content-based multimedia information retrieval: State of the art and challenges. *ACM Trans. Multimedia Comput. Commun. Appl.*, 2(1):1–19, 2006.

[148] Florence Tushabe and Michael.H.F. Wilkinson. Content-based image retrieval using combined 2D attribute pattern spectra. In *Advances in Multilingual and Multimodal Information Retrieval*, volume 5152 of *Lecture Notes in Computer Science*, pages 554–561, 2008.

[149] S. Brandt, J. Laaksonen, and E. Oja. Statistical shape features in content-based image retrieval. In *15th International Conference on Pattern Recognition*, volume 2, pages 1062 –1065, 2000.

[150] K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2(6):559–572, 1901.

[151] Xiang Bai, Wenyu Liu, and Zhuowen Tu. Integrating contour and skeleton for shape classification. In *IEEE 12th International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 360 –367, 2009.

[152] A. Vedaldi and B. Fulkerson. Vlfeat: An open and portable library of computer vision algorithms, 2008.

[153] Boris G. Mirkin. *Mathematical classification and clustering*. Kluwer Academic Press., 1996.

[154] Nancy Chinchor. MUC-4 evaluation metrics. In *Proceedings of the 4th conference on Message Understanding*, pages 22–29, 1992.

[155] Yujing Zeng, Jianshan Tang, Javier Garcia-Frias, and Guang R. Gao. An adaptive meta-clustering approach: Combining the information from different clustering results. In *Proceedings of the IEEE Computer Society Conference on Bioinformatics*, pages 276–287, 2002.

[156] Andrew Rosenberg and Julia Hirschberg. V-measure: A conditional entropy-based external cluster evaluation measure. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 410–420. Association for Computational Linguistics, 2007.

[157] Marina Meila. Comparing clusterings-an information based distance. *Journal of Multivariate Analysis*, 98(5):873 – 895, 2007.

[158] Allan P. White and Wei Zhong Liu. Technical note: Bias in information-based measures in decision tree induction. *Machine Learning*, 15(3):321–329, 1994.

[159] Chunyuan Li, Xinge You, A. Ben Hamza, Wu Zeng, and Long Zhou. Distinctive parts for shape classification. In *International Conference on Wavelet Analysis and Pattern Recognition (ICWAPR)*, pages 97 –102, 2011.

[160] Y. Li, J. Zhu, and F.L. Li. A hierarchical shape tree for shape classification. In *25th International Conference of Image and Vision Computing New Zealand (IVCNZ)*, pages 1 –6, 2010.

[161] Bingbing Ni. *Learning with Contexts*. PhD thesis, National University of Singapore, 2010.

[162] Haibin Ling and D.W. Jacobs. Shape classification using the inner-distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(2):286 –299, feb. 2007.

[163] G.H. Granlund. Fourier preprocessing for hand print character recognition. *IEEE Transactions on Computers*, C-21(2):195–201, Feb 1972.

[164] Dengsheng Zhang and Guojun Lu. Shape-based image retrieval using generic fourier descriptor. *Signal Processing: Image Communication*, 17(10):825 – 848, 2002.

[165] Rafael C Gonzalez, Richard Eugene Woods, and Steven L Eddins. *Digital image processing using MATLAB*. Pearson Education India, 2004.

[166] Antonin Chambolle. Total variation minimization and a class of binary MRF models. In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, volume 3757 of *Lecture Notes in Computer Science*, pages 136–152. Springer Berlin Heidelberg, 2005.

[167] B. Leibe and B. Schiele. Analyzing appearance and contour based methods for object categorization. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 409–15, 2003.

[168] Rong-Xiang Hu, Wei Jia, Yang Zhao, and Jie Gui. Perceptually motivated morphological strategies for shape retrieval. *Pattern Recognition*, 45(9):3222 – 3230, 2012.

[169] Junwei Wang, Xiang Bai, Xinge You, Wenyu Liu, and Longin Jan Latecki. Shape matching and classification using height functions. *Pattern Recognition Letters*, 33(2):134 – 143, 2012.

[170] J. Kittler and J. Illingworth. Minimum error thresholding. *Pattern Recognition*, 19(1):41 – 47, 1986.

[171] Judith M. S. Prewitt and Mortimer L. Mendelsohn. The analysis of cell images. *Annals of the New York Academy of Sciences*, 128(3):1035–1053, 1966.

[172] Wen-Hsiang Tsai. Moment-preserving thresolding: A new approach. *Computer Vision, Graphics, and Image Processing*, 29(3):377 – 393, 1985.

[173] N. Otsu. A threshold selection method from gray-level histograms. *Automatica*, 11(285-296):23–27, 1975.

[174] A. Rosenfeld and P. de la Torre. Histogram concavity analysis as an aid in threshold selection. *IEEE Transactions on Systems, Man and Cybernetics*, 13(2):231–235, 1983.

[175] JN Kapur, P.K. Sahoo, and AKC Wong. A new method for gray-level picture thresholding using the entropy of the histogram. *Computer vision, graphics, and image processing*, 29(3):273–285, 1985.

[176] C.A. Glasbey. An analysis of histogram-based thresholding algorithms. *CVGIP: Graphical Models and Image Processing*, 55(6):532 – 537, 1993.

[177] W. Doyle. Operations useful for similarity-invariant pattern recognition. *Journal of the Association for Computing Machinery*, 9(2):259–267, 1962.

[178] Emanuel Parzen. On estimation of a probability density function and mode. *Ann. Math. Statist.*, 33(3):1065–1076, 1962.

[179] Adrian W Bowman and Adelchi Azzalini. *Applied smoothing techniques for data analysis*. Clarendon Press, 2004.

[180] Z. I. Botev, J. F. Grotowski, and D. P. Kroese. Kernel density estimation via diffusion. *Ann. Statist.*, 38(5):2916–2957, 2010.

[181] V. A. Epanechnikov. Non-parametric estimation of a multivariate probability density. *Theory of Probability & Its Applications*, 14(1):153–158, 1969.

[182] Matt P Wand and M Chris Jones. *Kernel smoothing*. CRC Press, 1994.

[183] Bernard W Silverman. *Density estimation for statistics and data analysis*, volume 26. CRC press, 1986.

[184] Tadmeri Narayan Vikram, Marko Tscherepanow, and Britta Wrede. A saliency map based on sampling an image into random rectangular regions of interest. *Pattern Recognition*, 45(9):3114 – 3124, 2012.

[185] S. Chikkerur, T. Serre, C. Tan, and T. Poggio. What and where: A bayesian inference theory of attention. *Vision research*, 50(22):2233–2247, 2010.

[186] Stephen M. Pizer, E. Philip Amburn, John D. Austin, Robert Cromartie, Ari Geselowitz, Trey Greer, Bart ter Haar Romeny, John B. Zimmerman, and Karel Zuiderveld. Adaptive histogram equalization and its variations. *Computer Vision, Graphics, and Image Processing*, 39(3):355 – 368, 1987.

[187] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145–175, 2001.

[188] 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 *Proceedings of the IEEE Workshop on Generative-Model Based Vision, CVPR*, 2004.

[189] Lamberto Ballan, Marco Bertini, Alberto Del Bimbo, Andrea M. Serain, Giuseppe Serra, and Benito F. Zaccone. Combining generative and discriminative models for classifying social images from 101 object categories. In *Proc. of International Conference on Pattern Recognition (ICPR)*, 2012.

[190] Jan C. van Gemert, Jan-Mark Geusebroek, Cor J. Veenman, and Arnold W.M. Smeulders. Kernel codebooks for scene categorization. In *European Conference on Computer Vision*, volume 5304 of *Lecture Notes in Computer Science*, pages 696–709, 2008.

[191] Hao Zhang, A.C. Berg, M. Maire, and J. Malik. SVM-KNN: Discriminative nearest neighbor classification for visual category recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2126–2136, 2006.

[192] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. Technical Report 7694, California Institute of Technology, 2007.

[193] O. Boiman, E. Shechtman, and M. Irani. In defense of nearest-neighbor based image classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[194] P. Jain, B. Kulis, and K. Grauman. Fast image search for learned metrics. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[195] Yangmuzi Zhang, Zhuolin Jiang, and L.S. Davis. Learning structured low-rank representations for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 676–683, 2013.

[196] A. Shaban, H.R. Rabiee, M. Farajtabar, and M. Ghazvininejad. From local similarity to global coding: An application to image classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2794–2801, 2013.

[197] Abhishek Kumar, Alexandru Niculescu-mizil, Koray Kavukcuoglu, and Hal Daume. A binary classification framework for two-stage multiple kernel learning. In *Proceedings of the 29th International Conference on Machine Learning*, pages 1295–1302, 2012.

[198] C. Kanan and G. Cottrell. Robust classification of objects, faces, and flowers using natural image statistics. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2472–2479, 2010.

[199] G.L. Oliveira, E.R. Nascimento, A.W. Vieira, and M.F.M. Campos. Sparse spatial coding: A novel approach for efficient and accurate object recognition. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2592–2598, 2012.

[200] Mu Qiao and Jia Li. Distance-based mixture modeling for classification via hypothetical local mapping. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, page Online Version (yet to be issued), 2015.

[201] L. Seidenari, G. Serra, A.D. Bagdanov, and A. Del Bimbo. Local pyramidal descriptors for image recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(5):1033–1040, 2014.

[202] Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision*, volume 8689 of *Lecture Notes in Computer Science*, pages 818–833, 2014.

[203] Na Tong, Huchuan Lu, Lihe Zhang, and Xiang Ruan. Saliency detection with multi-scale superpixels. *IEEE Signal Processing Letters*, 21(9):1035–1039, 2014.

[204] Yulin Xie and Huchuan Lu. Visual saliency detection based on bayesian model. In *IEEE International Conference on Image Processing*, pages 645–648, 2011.

[205] Xiaohui Li, Huchuan Lu, Lihe Zhang, Xiang Ruan, and Ming-Hsuan Yang. Saliency detection via dense and sparse reconstruction. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2976–2983, 2013.

[206] Bowen Jiang, Lihe Zhang, Huchuan Lu, Chuan Yang, and Ming-Hsuan Yang. Saliency detection via absorbing Markov chain. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1665–1672, 2013.

[207] Vance A Tucker. The deep fovea, sideways vision and spiral flight paths in raptors. *Journal of Experimental Biology*, 203(24):3745–3754, 2000.

[208] Allan W Snyder and William H Miller. Telephoto lens system of falconiform eyes. *Nature*, 275:127 – 129, 1978.

[209] BC Richmond. *A Practical Guide to USB 3.0 for Vision applications*. Point Grey Research, Inc, 12051 Riverside Way, Feb 2013.

[210] P. KaewTraKulPong and R. Bowden. An improved adaptive background mixture model for real-time tracking with shadow detection. In *Video-Based Surveillance Systems*, pages 135–144, 2002.

[211] Chris Stauffer and W Eric L Grimson. Adaptive background mixture models for real-time tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, 1999.

[212] Y.T. Chan, A.G.C. Hu, and J.B. Plant. A Kalman filter based tracking scheme with input estimation. *IEEE Transactions on Aerospace and Electronic Systems*, 15(2):237–244, 1979.

[213] Edward Rosten and Tom Drummond. Fusing points and lines for high performance tracking. In *IEEE International Conference on Computer Vision*, volume 2, pages 1508–1515, 2005.

[214] J. Barroso, E.L. Dagless, A. Rafael, and J. Bulas-Cruz. Number plate reading using computer vision. In *Proceedings of the IEEE International Symposium on Industrial Electronics*, volume 3, pages 761–766, 1997.

[215] Clemens Arth, F. Limberger, and H. Bischof. Real-time license plate recognition on an embedded DSP-platform. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[216] P.F. Felzenszwalb, R.B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010.

[217] Li Congcong, A. Kowdle, A. Saxena, and Chen Tsuhan. Toward holistic scene understanding: Feedback enabled cascaded classification models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(7):1394–1408, 2012.

Appendix: Publication List

Journal Papers

B. Ramesh, C. Xiang and T. H. Lee, "Shape classification using invariant features and contextual information in the bag-of-words model", *Patt. Recog.*, vol. 48, no. 3, pp. 894-906, Mar 2015.

B. Ramesh, C. Xiang and T. H. Lee, "Multiple Object Cues for High Performance Vector Quantization", submitted to *Pattern Recognition*, 2015.

B. Ramesh, C. Xiang and T. H. Lee, "Cue-based Unseen Object Categorization using Optimized Visual Dictionaries", submitted to *Computer Vision and Image Understanding*, 2015.

Conference Papers

B. Ramesh, C. Xiang and T. H. Lee, "Real-time Shape Classification Using Biologically Inspired Invariant Features", *Proc. IEEE Symposium on Computational Intelligence for Multimedia, Signal and Vision Processing (CIMSIVP)*, pp. 1-8, Dec 2014.

L. Lin, B. Ramesh, and C. Xiang, "Biologically Inspired Composite Vision System for Multiple Depth-of-field Vehicle Tracking and Speed Detection", *Com-*

puter Vision - ACCV 2014 Workshops, volume 9008 of *Lecture Notes in Computer Science* , pp. 473-486, Springer International Publishing, 2015.