Image Classification Using Appearance Based Features

Dina Masri, Zeyar Aung, Wei Lee Woon
Electrical Engineering and Computer Science,
Masdar Institute of Science and Technology,
P.O. Box 54224, Abu Dhabi, UAE.
Email: {dmasri,zaung,wwoon}@masdar.ac.ae

***Abstract*—In this paper, a small set of features based on local appearance and texture is applied to the task of image recognition and classification. These features are used to train and subsequently test three different machine learning techniques, namely k-Nearest Neighbors (K-NN), Support Vector Machines (SVM) and Ensemble Learning (Bagging). A case study on a publicly available object classification dataset was conductor from which it was concluded that, while simple, the proposed approach was able to produce extremely high classification accuracies.**

***Keywords*—Image Classification, Features Extraction, Texture Analysis, Edge Detection, Object Recognition.**

I. Introduction

The detection and classification of objects in images is a challenging problem in computer vision and underlies such commonly encountered applications as the photo tagging features in social media platforms like Facebook and Instagram, Google's Image Search and intelligent video surveillance systems. In addition, it is a critical first step in a wide range of higher-level vision tasks including activity or event recognition and scene understanding.

While many image recognition methods have been proposed (discussed at greater length in Section II-A), they often require highly engineered feature sets which can be computationally and conceptually complex. A more recent development is Deep Learning, which is based on neural networks with multiple hidden layers which perform automatic feature learning and extraction. While these methods can be extremely accurate, they have relatively high computational and data requirements and can sometimes be an "overkill" in the context of simple tasks where only limited computing resources are available or required.

The main contributions of this paper are as follows:

1) A *review* of image classification methods and applications.
2) A *novel methodology* for image classification which uses a set of local appearance and texture based features combined with a classification algorithm.
3) A *case study* using the publicly available NEC Animal images data set [1].

II. Background and Related Work

A. General Approach and Feature Extraction

Image classification is the task of recognizing and labeling images or objects present within images based on the properties, appearance or textures associated with these images. Object detection generally consists of two different variants; object instance recognition, and object class recognition [21]. The first one is a supervised classification task which aims at identifying previously classified classes or images. The second one is a category level task, where previously unseen objects are grouped according to predefined categories. The second category is far more challenging and difficult than the first, given that many variants may be present within the same category, which are caused by different colors, textures, imaging conditions and others.

The first step in classifying an image or an object is to extract appropriate features/descriptors. There are three main approaches [19], [17], [21]: Model, Shape and Appearance based features. Briefly, model based features seek to approximate objects using low level primitives such as boxes, cones and the like. Shape based features try to extract the boundaries of objects using edge detection and contour finding methods. The third approach (Appearance based) is by far the most common will now be discussed in greater detail.

Appearance Based Methods

As mentioned, this is the most widely used of the three approaches. These methods involve the extraction of visual features which bridge the gap between low level appearance features and higher level semantic features. This model can be further divided into two main sub-models depending on the types of appearance features used, these are: Appearance Models with Local Features, and Appearance Models with Global Features. One other model that mostly combines between the concepts of global and local features and that is Appearance models with textural features.

- *Appearance based Models with Local Features*
  A local feature is a property of a small part or region of an image which describes the object's projection to the camera [17], [20]. This type of features can either be on pixel- or patch-level [21]. Pixel-level features are calculated for each pixel (either gray level or RGB vectors), where descriptors like pixel intensities, colors, textures, edges are employed. However, such features are sensitive to changes in illumination levels, scale and noise.
  Patch-level features address this issue by grouping pixels into regions and using these to generate more complex descriptors. There are two main variants: corner- and region-based. Edge detection is the main method used, where local gradients in grey level intensities are used

978-1-4673-8511-4/15 $31.00 ©2015 IEEE

to detect object boundaries. Region based features detect local "blobs" characterized by uniform brightness levels. Examples of such descriptors are scale invariant feature transform SIFT, gradient location-orientation histograms (GLOH), sometimes also called extended SIFT, Histograms of Oriented Gradients (HOG), Locally Binary Patterns [17]. SIFT determines highly repetitive interest points at an estimated scale (location and scale in-variant), and is sometimes known as the difference of Gaussian (DoG) detector.

Another popular region based local feature scheme is "Speeded-Up Robust Features" (SURF)[2] which is a novel scale- and rotation-invariant detector and descriptor. It outperforms previously proposed schemes with respect to repeatability, distinctiveness, and robustness. Surf approach is mainly for interest point detection and it uses a very basic Hessian-matrix approximation.

- *Appearance based Models with Global Features*
  This type of feature uses global information about the whole scene or image [20]. Color histograms and the variants thereof are examples of these features. This kind of feature can work well for images with distinctive colors, as long as the interest is in finding the overall composition of the image in question and not a specific object. As such, Global features are commonly used to describe the whole image (including the object of interest) which can support the object recognition task. Other widely used global feature approaches include Principal Component Analysis (PCA), Independent Component Analysis (ICA), or Zero Component Analysis (ZCA or whitening) [20], [17]. These methods use projections of the image grayscale values onto small sets of descriptive basis vectors such as the principal components (in the case of PCA).
  Compared to Local Features (geometry and sub-image based), Global features can produce markedly improved classification results, especially for natural images [20]. However, when the data under study has clean backgrounds, and where objects can be easily demarcated using segmentation algorithms, they might perform poorly compared to local features, where image clutter and occlusions can pose a major problems for global features.
- *Appearance based Models with Textural Features*
  In this model, different statistical properties, using grey-tone spatial dependencies and regional intensity levels of image pixels, for the image as a whole are used [7]. Here, regional, geometric and spatial information are taken into account, while considering the whole image in the process. The main technique used for this type of feature model is called Grey-Level Co-occurrence Matrix (GLCM). GLCM comprise the frequencies of occurrence of a certain intensity level in a specific region. Many statistical calculations can be performed on the resulting matrix like Energy calculation and homogeneity level of the image texture. More details will be provided in section IV.

After pre-processing and feature extraction, classification techniques can then be applied to dintinguish between the different object classes that may be present in an image. In the following section, this subsequent stage is discussed in greater detail.

B. Object Classification

Image classification and object recognition has been addressed elaborately in literature. In [19] a novel shape descriptor called 'chordiogram' was introduced, it is based on the idea of holism. Also, a shape based object detection was proposed, where a method called Boundary Structure Segmentation (BoSS) was developed. Based on a novel holistic shape descriptor, this method relates object detection to ground segmentation, where it selects a foreground region with properties like Similarity in Shape,and Perceptual Saliency, based on configural cues of salient contours, color and texture coherence, and small perimeter prior. Using BoSS, simultaneous shape matching and segmentation is done simultaneously and as a result, enable holistic shape-based object detection in cluttered scenes.

In [5] SVM was used for multiclass classification of airborne sensor data, and is evaluated against a series of classifiers that are widely used in remote sensing, like decision trees, ANNs, and discriminant analysis. The input feature space was a per-pixel appearance based model. The SVM classifier outperformed the other classifiers using a single binary based SVM model. In [15] a new clustering scheme, called Extremely Randomized Clustering (ERC) Forest, which is very efficient for vector quantization of local visual information was used to quantize patch-based local features model. The proposed random based decision tree model was found to be robust to background clutter and provides relatively clean foreground class segmentations. Moreover, the ERC was augmented with visual search mechanisms so that only useful visual information is processed where saliency maps were used. Ensemble learning with ANNs being the week learners were used for image classification in [6], where pixel-based approach for agricultural remote sensed images was used.

Unsupervised learning techniques such as ICA [12] have also been employed for image classification. ICA models the observed data as a mixture of several statistically independent classes, where the observations are assumed to be the linear combinations of class dependent random variables with non-Gaussian distributions. This was used for texture analysis of images.

There are many such approaches to the task of image classification and object recognition; however, at present and most active research direction is the use of Deep learning and Convolutional Neural Networks (CNN) along with the global feature model [8], [14], [4], [11]. In [11] face recognition was performed using a system which combines local image sampling, a self-organizing map (SOM) neural network, and a convolutional neural network. The SOM was used to perform quantization of the image samples, dimensionality reduction and invariance to minor changes in the image. The convolutional neural network provides for partial invariance to translation, rotation, scale, and deformation. The input feature space are sampled pixel intensity values from local windows in the images.

Deep convolutional networks were used in [14] and [4]. Human competitive results were produced in [14] which is impressive, and the recognition rate drastically exceeded the state of the art at the time. An interesting approach was presented in [4] which leveraged the temporal nature of the

data by using coherence as a supervisory signal for unlabeled data. The intuition behind this approach was that consecutive images in a video sequence ar likely to be very similar. A deep convolutional network architecture was used, and different datasets were studied, one of which is the NEC animal dataset that is the focus of this study. However in their approach, a video was created of the animal images in different poses. The performance of this approach was compared against K-NN and SVM. For the NEC dataset, the average accuracy achieved for all the animals was about 78.67%, which is impressive for unsupervised and unlabeled input data.

In general, image recognition is faced with a lot of challenges, most of which are related to robustness, computational-complexity- and scalability. Also, the choice of recognition and classification method used is highly dependent on the image dataset and quality, as well as on the available computational resources. So, although deep neural networks can potentially provide the highest classification accuracy, available computating resources and data may not always be sufficient. This is why in this work, a conscious choice was made to use simple classifiers and features.

III. Test Case: NEC Animal Dataset Classification

In this section, a case study on image feature extraction and classification is conducted. The dataset under study is NEC animal dataset provided by NEC laboratories America [1]. It is a publicly available and has already been the subject of a number of other studies [13], [18]. Fig. 1 portrays a sample of the dataset. As mentioned earlier, the dataset consists of about 5000 images of 60 animals, where each animal has about 72 different poses.

The following sub-sections discuss the feature extraction process as well as the classification methods used.

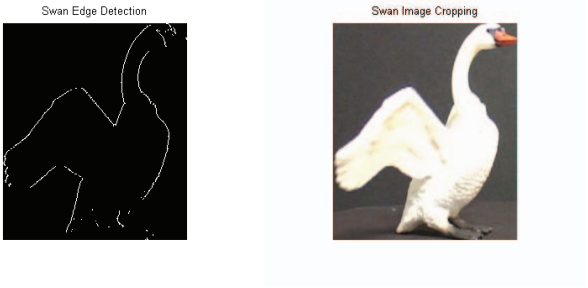
A. Feature Extraction

A mix of Appearance based local and textural features were used. Nine Different features were extracted using four different Matlab Image Processing Toolbox functionalities, namely Edge Detection, Color Histogram, Binarization and binary regions properties, and GLCM matrix statistics. Table I summarizes the techniques used, features extracted and the corresponding matlab functions.

Fig. 1: Sample of NEC Animal Dataset

TABLE I: Image Feature Extraction Methods

Method/Functionality	Matlab Function	Extracted Features
Edge Detection	*edge*	Animal Hight Animal Area of bounding box
Binarization	*im2bw* and *regionprops*	Animal Area Relative area to the bounding box
Color Histogram	*imhist*	Number of unique colors
GLCM	*graycomatrix* and *graycoprops*	Texture Contrast Texture Correlation Texture Energy Texture homogeneity



(a) Edge Detection (b) Cropping

Fig. 2: Edge Detection and Cropping of the Swan Image

Edge Detection.

An edge is a region or line in an image where there is a rapid change in image intensity and which is hence usually associated with the boundaries of an object in the image. The detection of edges is often performed using an approximation of the first derivative of the gray level values, though many enhancements to this basic formulation have been proposed (for e.g. [16]). Different edge finding methods are provided by Matlab like Prewitt, Sobel, and Canny. The one chosen for this work is the Prewitt edge detector. The threshold of the gradient intensity difference was set empirically, where different images with extreme intensity values (high and low) and colors where tested, and a compromise was reached which maximized objection detection while minimizing false positives. In addition, it was noticed that the tables which supported the animals tended to produce a large number of false positives. This problem was solved by only considering vertical edges.

Two main features were extracted from the images using edge detection, the *height* and the *area of the bounding box* of the animal (height $\times$ width), where the highest and lowest X and Y axis indexes of the image were obtained from the binary image that results from the edge detection function '*edge*'. In fig. 2 (a) the result of edge detection for the swan image is presented. Using the height and width information, the image was cropped, and the rest of the features were extracted from the cropped images. This is because it would be more focused on the object itself rather than the background, especially for the textural analysis.

Binarization. The cropped images were binarized according to an intensity based relative threshold which was set empirically. In the resulting binary image, white regions (1's) are areas where the original image intensity exceeded the

threshold, and are assumed to correspond to objects in the image (this works well in this context though this stage can be replaced with more advanced segmentation algorithms as and when this is required by the dataset/application). Fig. 3 shows the binarized swan image. It can be seen that the swan object was perfectly detected and replaced with a white area. Using the Binary region properties provided by matlab, many information about the binarized image can be extracted; like the number of white blobs in the region, the perimeter of each blob, the centroid, the area, and many others. However, since we are interested in detecting the object as a whole, the property of area was taken into consideration. This property returns the overall area of the white regions in a binary image in $pixel^2$.

Two features from the area property were extracted; The surface area of the animal and the relative area of the animal with respect to the area of the bounding box extracted previously.

Color Histogram. Color histograms represent the frequency of occurrence of all color indexes in an image with different intensities. The feature extracted from this property is the number of unique colors of the animal. The intuition behind this is that there are animals with very few colors and others very colorful. So it was assumed that this feature can be a decision boundary advantage for animals of close regional features.

GLCM. GLCM is a recurrence matrix that represents how many times a certain grey scale intensity is repeated in neighboring pixels with certain offset [7]. Fig. 4 which was taken from mathworks website shows in a simplified manner how the matrix is built. The matrix on the left is the grey scale intensity matrix per-pixel, and to the left is the GLCM being filled with recurrence values.

After constructing the co-occurrence matrix, four different statistical properties can be calculated; these are Contrast, Correlation, Energy, and Homogeneity [7]. Contrast property is a measure of the intensity contrast between a pixel and its neighbor, this indicate the amount of local variations present in the image. Correlation property is a measure of grey-tone linear dependencies. In other words it is a measure of how correlated a pixel is to its neighbor over the whole image, do for a constant image or image with the same texture would produce 100% correlation. Energy feature is simply the sum of squared elements of the GLCM, meaning that constant images would produce 100% energy since the values in the GLCM would be very high. Finally, Homogeneity measures measures the closeness of the distribution of elements in the image. It is a second moment feature, and high homogeneity means that there are very few dominant gray-tone transitions in the GLCM.

These textural features are assumed to provide important and informative features about the texture of the animal fur or feather, and its colors. These features along with the other local spacial and regional features should provide enough information about each animal, in order for the classifier to be able to distinguish successfully between animal classes. Fig. 5 shows an attempt to study the extracted features by checking the distribution of these classes against two of the features. Fig. 5 (a) portrays the scatter plot of of the animal height against

Fig. 3: Binarized Swan Image

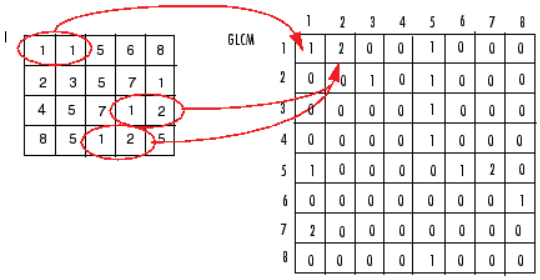


Fig. 4: GLCM Building Process

surface area for three white animals with similar textures and colors; The two types of white ducks and the white chicken. The distribution of features is a bit noisy however one can see clearly that these type of features would provide useful information for any classifier.

On the other hand, in fig. 5 (b), the scatter plot of Homogeneity against number of distinct colors for another three animals is displayed. These animal; The rooster, brown chicken and colored chicken, had very close spacial distributions but very different textures and colors. Again, the usefulness of such features for the classification can be seen from the three class distributions.

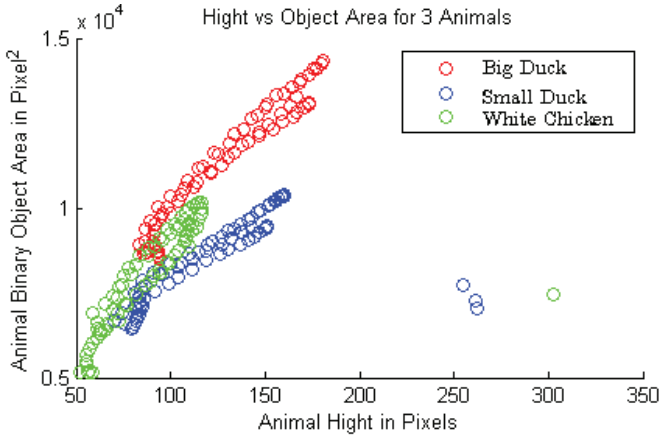
B. Image Classification

After the nine features were extracted, feature selection was performed using the wrapper based technique [9], which evaluates different feature subsets in terms of classification accuracy using a number of different classifiers. This resulted in the exclusion of the Relative Animal Surface area.

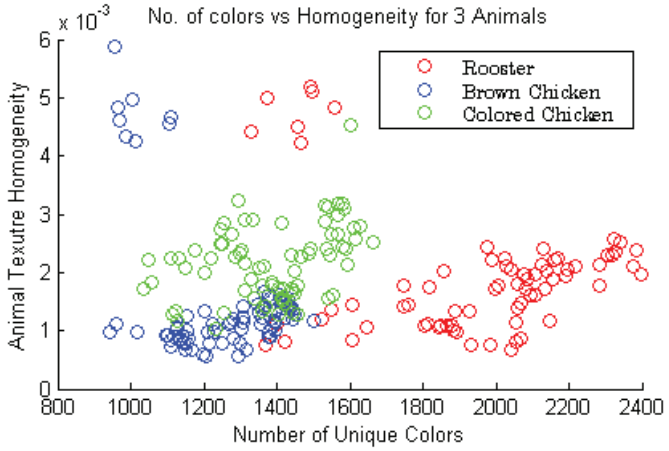
Three different classifiers were used to test the validity of the selected features; K-NN, SVM, and Ensemble Decision Trees (Bagging). The K-NN implementation provided by Matlab was used while for SVM, the toolbox provided by [10] was

TABLE II: Classification Results

Model	Model Parameters	Number of Classes	CV Accuracy	Test Accuracy
Weighted KNN	k = 1	10 Classes	0.9716	0.984
	k = 1	15 Classes	0.9794	0.9892
	k = 1	30 Classes	0.936	0.9099
	k = 1	60 Classes	0.8912	0.8914
SVM	γ = 0.1, c = 1000	10 Classes	0.99	**1**
	γ = 0.2, c = 1000	15 Classes	**0.9924**	**1**
	γ = 0.5, c = 1000	30 Classes	0.982	0.997
Bagging	no. learners = 100	10 Classes	0.9820	0.996
	no. learners = 100	15 Classes	0.9772	0.9756
	no. learners = 150	30 Classes	0.9227	0.9376
	no. learners = 150	60 Classes	0.9076	0.9032



(a) Hight Vs Binary Surface Area Scatter Plot



(b) Homogeneity Vs Number of Distinct Colors Scatter Plot

Fig. 5: Scatter Plots for different features and animals

used as it supported built in multi-class Classification. RBF kernel was used for it is the most used kernel type for SVM. For the Ensemble of Decision Trees, Bagging topology, which is matlab's default tree building block structure, was used.

Testing was conducted using 10, 15, 30, and 60 (full dataset) animal classes. 10-Fold stratified cross validation was used for parameter tuning while 33% of the full data was set aside to calculate the final (validation) accuracy. In the following section, the classification results of the three techniques were reported.

C. Results and Discussion

The classification results of the different testing/training schemes, classes, and test percentages are provided in Table II. Two things worth mentioning is that for SVM, only up to 30 different animal classes were tested due to the lack of proper computational equipment and lack of enough time, where it was taking the machine sometimes more than one hours to perform one training attempt.

Looking at the results as a whole, one can easily see that regardless of the technique, all the classification results were exceptional, even for the whole data set consisting of 60 different animal classes, with animals that might seem very similar to each other. Naturally, as the number of classes increase, the classification error slightly increased. The most impressive results achieved using the SVM classifier, where 100% accuracy level was achieved on two occasions (see Table ref:results). Unfortunately due to time and resource constraints we were unable to test the SVM classifier on the full (60-class) dataset, there is no reason to suspect that this performance level will be significantly decreased as no such effect is experienced when using the other two classifiers.

Ensemble learning on the other hand produced very similar results to K-NN, which confirms the findings in [3] that K-NN can perform very well and is comparable to the state of the art classification techniques like ensemble learning when used with strong local features such as those used in this study. It is important to mention that for SVM and K-NN, normalizing the features to values between 0 and 1 substantially increased

the resulting accuracy; for example, in the K-NN case, in some cases it increased from 70% to 99%. This implies that, unlike decision trees, both SVM and K-NN are sensitive to highly variable input feature scales.

IV. Conclusion

This paper focuses on the challenge of image classification and object recognition in particular. As part of this work, a review of different feature models was conducted and forms part of the contribution of this paper.

The proposed method was tested via a case study on the publicly available NEC Animal dataset, were different local and textural features were extracted from the images. These features included descriptors such as the height and area of the bounding box, the surface area of the animals, as well as the number of colors. Three different classification techniques were tested with different numbers of classes. From the classification results one can conclude that sometimes, simple local features for datasets like the one under study can be the best solution, and even a simple classifier like the K-NN can produce very high accuracies. Another conclusion is that SVM is superior to the other two for this classification task and in some cases had the ability to perfectly classify the test data, though this success was tempered by its extremely long training times. A final conclusion is that K-NN and SVm are both very sensitive to highly variant feature scales, and normalization is crucial for such cases in order to get good performance results.

References

[1] NEC Laboratories America. *NEC Animal Dataset*, 2009.

[2] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):346–359, 2008.

[3] Oren Boiman, Eli Shechtman, and Michal Irani. In defense of nearest-neighbor based image classification. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[4] Dan Ciresan, Ueli Meier, and Jürgen Schmidhuber. Multi-column deep neural networks for image classification. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3642–3649. IEEE, 2012.

[5] Giles M Foody and Ajay Mathur. A relative evaluation of multiclass image classification by support vector machines. *Geoscience and Remote Sensing, IEEE Transactions on*, 42(6):1335–1343, 2004.

[6] Giorgio Giacinto and Fabio Roli. Design of effective neural network ensembles for image classification purposes. *Image and Vision Computing*, 19(9):699–707, 2001.

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

[8] Geoffrey E Hinton. To recognize shapes, first learn to generate images. *Progress in brain research*, 165:535–547, 2007.

[9] Ron Kohavi and George H John. Wrappers for feature subset selection. *Artificial intelligence*, 97(1):273–324, 1997.

[10] F. Lauer and Y. Guermeur. MSVMpack: a multi-class support vector machine package. *Journal of Machine Learning Research*, 12:2269–2272, 2011.

[11] Steve Lawrence, C Lee Giles, Ah Chung Tsoi, and Andrew D Back. Face recognition: A convolutional neural-network approach. *Neural Networks, IEEE Transactions on*, 8(1):98–113, 1997.

[12] Te-Won Lee and Michael S Lewicki. Unsupervised image classification, segmentation, and enhancement using ica mixture models. *Image Processing, IEEE Transactions on*, 11(3):270–279, 2002.

[13] Prajowal Manandhar, Zeyar Aung, Wei Lee Woon, and Prashanth Marpu. Random forest ensemble learning for object recognition using rgb features along object edge. In *ICCIT 2015 The Fourth International Conference on Communications and Information Technology*, Dubai, UAE, 2015. IEEE.

[14] Hossein Mobahi, Ronan Collobert, and Jason Weston. Deep learning from temporal coherence in video. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 737–744. ACM, 2009.

[15] Frank Moosmann, Eric Nowak, and Frederic Jurie. Randomized clustering forests for image classification. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(9):1632–1646, 2008.

[16] Bijay Neupane, Zeyar Aung, and Wei Lee Woon. A new image edge detection method using quality-based clustering. In *Proceedings of the 10th IASTED International Conference on Visualization, Imaging, and Image Processing*, pages 20–26, 2012.

[17] Peter M Roth and Martin Winter. Survey of appearance-based methods for object recognition. *Inst. for Computer Graphics and Vision, Graz University of Technology, Austria, Technical Report ICGTR0108 (ICG-TR-01/08)*, 2008.

[18] Sachin Sharma, Dharmesh Shah, Rishikesh Bhavsar, Bhavesh Jaiswal, and Kishor Bamniya. Automated detection of animals in context to indian scenario. In *ISMS 2014 - The Fifth International Conference on Intelligent Systems, Modelling and Simulation*, Langkawi, Malaysia, 2014. IEEE.

[19] Alexander Toshev, Ben Taskar, and Kostas Daniilidis. Shape-based object detection via boundary structure segmentation. *International journal of computer vision*, 99(2):123–146, 2012.

[20] Tinne Tuytelaars and Krystian Mikolajczyk. Local invariant feature detectors: a survey. *Foundations and Trends® in Computer Graphics and Vision*, 3(3):177–280, 2008.

[21] Xin Zhang, Yee-Hong Yang, Zhiguang Han, Hui Wang, and Chao Gao. Object class detection: A survey. *ACM Computing Surveys (CSUR)*, 46(1):10, 2013.