SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

Automatic Region-of-Interest Extraction in Low Depth-of-Field Images

By

Gholamreza Rafiee

A THESIS

SUBMITTED TO THE FACULTY OF SCIENCE, AGRICULTURE AND ENGINEERING

IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR THE DEGREE

DOCTOR OF PHILOSOPHY

SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

NEWCASTLE UNIVERSITY

UNITED KINGDOM

July 2013

NEWCASTLE UNIVERSITY

SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

I, *Gholamreza Rafiee,* confirm that this thesis and work presented in it are my own achievement.

I have read and understand the penalties associated with plagiarism.

Signature:

Date: July 2013

SUPERVISOR'S CERTIFICATE

This is to certify that the entitled thesis “Automatic Region-of-Interest Extraction in Low Depth-of-Field Images" has been prepared under my supervision at the school of Electrical and Electronic Engineering in Newcastle University for the degree of PhD in Computer Engineering – Image Processing/Computer Vision.

Signature:

Supervisor: Professor Satnam Dlay

Date: July 2013

Signature:

Student: Gholamreza Rafiee

Date: July 2013

ABSTRACT

Automatic extraction of focused regions from images with low depth-of-field (DOF) is a problem without an efficient solution yet. The capability of extracting focused regions can help to bridge the semantic gap by integrating image regions which are meaningfully relevant and generally do not exhibit uniform visual characteristics. There exist two main difficulties for extracting focused regions from low DOF images using high-frequency based techniques: computational complexity and performance.

A novel unsupervised segmentation approach based on ensemble clustering is proposed to extract the focused regions from low DOF images in two stages. The first stage is to cluster image blocks in a joint contrast-energy feature space into three constituent groups. To achieve this, we make use of a normal mixture-based model along with standard expectation-maximization (EM) algorithm at two consecutive levels of block size. To avoid the common problem of local optima experienced in many models, an ensemble EM clustering algorithm is proposed. As a result, relevant blocks, i.e., block-based region-of-interest (ROI), closely conforming to image objects are extracted.

In stage two, two different approaches have been developed to extract pixel-based ROI. In the first approach, a binary saliency map is constructed from the relevant blocks at the pixel level, which is based on difference of Gaussian (DOG) and binarization methods. Then, a set of morphological operations is employed to create the pixel-based ROI from the map. Experimental results demonstrate that the proposed approach achieves an average segmentation performance of 91.3% and is computationally 3 times faster than the best existing approach. In the second approach, a minimal graph cut is constructed by using the max-flow method and also by using object/background seeds provided by the ensemble clustering algorithm. Experimental results demonstrate an average segmentation performance of 91.7% and approximately 50% reduction of the average computational time by the proposed colour based approach compared with existing unsupervised approaches.

*For my wife, **Sara**, and my daughter **Dorsa***

ACKNOWLEDGMENTS

I would like to thank my supervisor Professor Satnam S. Dlay for all his guidance, support and encouragement. I have been privileged to know him and to work under his supervision. I am also grateful to Dr. Wai L. Woo for his guidance and excellent discussions on my research and special thanks to Mrs. Gillian Webber for her kindness, administrative support and help.

Above all, I would like to thank my wife and daughter for their incredible love, prayers, enthusiasm, and encouragement which have been instrumental in my academic achievement. I dedicate this thesis to them.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

CBIR	Content-Based Image Retrieval
DOF	Depth of Field
DOG	Difference of Gaussian
EM	Expectation Maximization
GMM	Gaussian Mixture Model
JPEG	Joint Photographic Expert Group
OOI	Object of Interest
PDF	Probability Density Function
PSF	Point-Spread Function
ROI	Region of Interest
RSM	Region Saliency Map

LIST OF AWARD AND PUBLICATIONS

Award

Awarded the international research scholarship from Azad University in Oxford.

Publications

1. G. Rafiee, Dlay S.S., Woo W.L.,, "Region-of-Interest Extraction in Low Depth of Field Images using Ensemble Clustering and Difference of Gaussian Approaches," *Pattern Recognition Journal*, vol. 46, pp. 2685-2699, 2013.

2. G. Rafiee, S. S. Dlay, and W. L. Woo, "Unsupervised Segmentation of Focused Regions in Images with Low Depth of Field," in *Multimedia and Expo (ICME), 2013 IEEE International Conference on*, San Jose, USA, in Press.

3. G. Rafiee, S. S. Dlay, and W. L. Woo, "Automatic Segmentation of Interest Regions in Low Depth of Field Images Using Ensemble Clustering and Graph Cut Optimization Approaches," in *Multimedia (ISM), 2012 IEEE International Symposium on*, California, USA, pp. 161-164.

4. G. Rafiee, S. S. Dlay, and W. L. Woo, "A Review of Content-Based Image Retrieval," in *Communication Systems Networks and Digital Signal Processing (CSNDSP), 2010 7th International Symposium on*, Newcastle, UK, pp. 775-779.

Chapter 1

1. INTRODUCTION

1.1 Region-of-Interest Extraction

With the development and widespread use of digital cameras, mobile phones with built-in cameras, and their ability to focus on any object within a taken photo, the number of Web images with focused regions is dramatically growing. Focused regions usually represent the visual attention objects and meaningful content of images. The recognition of visual attention objects plays an important role in many computer vision and multimedia applications. In photography, low depth-of-field (DOF) is an important technique commonly used to help viewers in understanding the depth information within a 2-D photo [1-2]. This technique usually produces a visual effect like object-in-focus, which means that only the objects/regions of interest (OOI/ROI) are in sharp focus, whereas background regions are typically blurred, being out-of-focus [3].

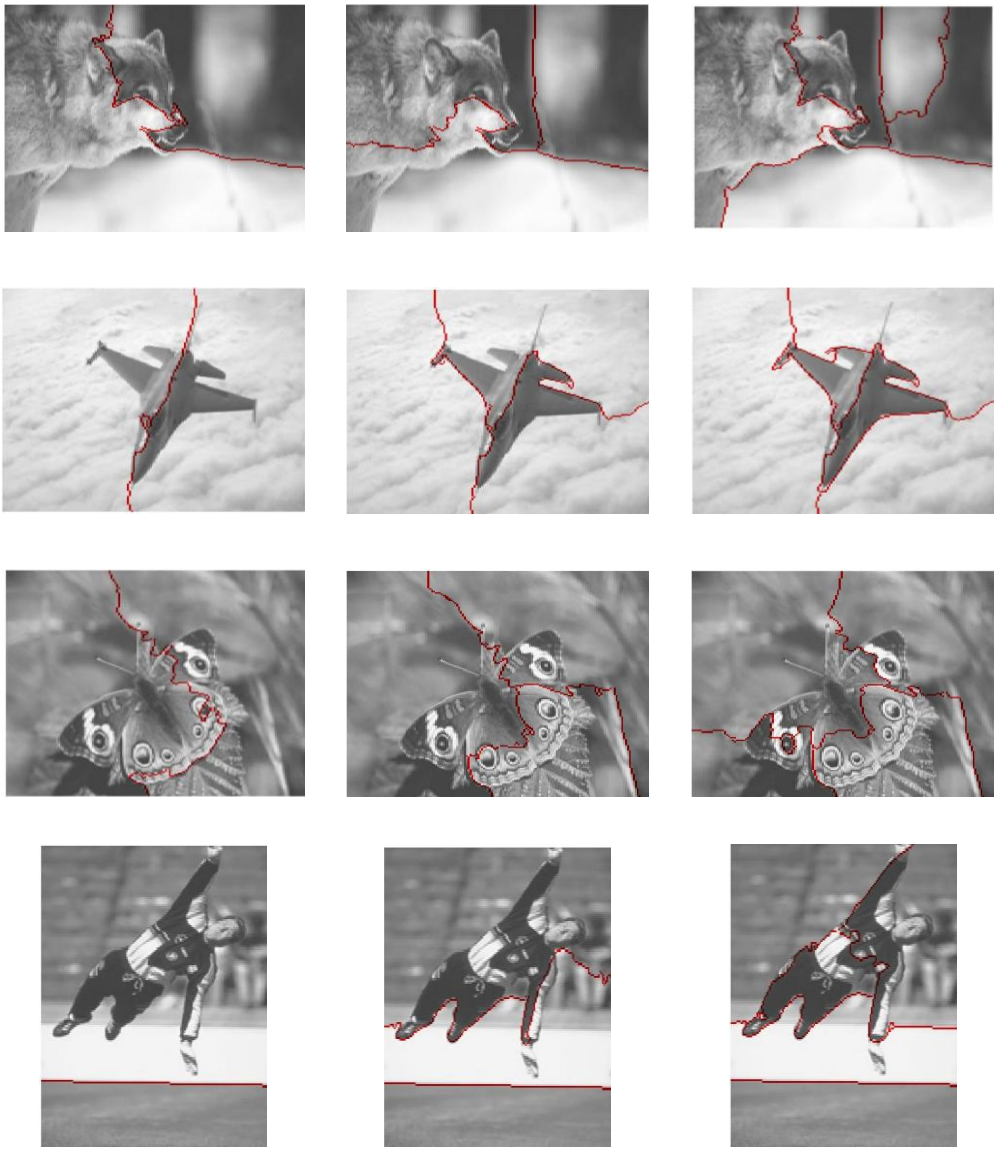
Low DOF images can be seen in different types of images such as sport, telephotos, close-up, and macro [3]. Figure 1.1 illustrates the different types of low DOF images.

Figure 1.1: Type of images with low DOF. (a) Sport. (b) Telephoto (i.e., long focal length). (c) Close-up (i.e., large aperture). (d) Macro.

Extracting focused objects [4], or more generally focused regions, in an image is a very important task for a wide range of image processing and computer vision applications such as image enhancement for digital cameras [3], target recognition [5], microscopic image analysis, image indexing [6], content-based image retrieval (CBIR) [7-10], object-based image retrieval [11], region-based image/video compression [12], image target searching [13], focus tracking in video frames in TV programs, and video target indexing [14].

1.2 Motivation and Challenges

Extracting automatically focused regions is a problem with no efficient solution yet. The capability of extracting focused regions can help to bridge the semantic gap [8], by integrating image regions which are meaningfully relevant and generally do not exhibit uniform visual characteristics. The main characteristic of low DOF images is that the focused regions normally have more high-frequency information than the defocused ones. This may be the only clue that one can utilise to automatically extract focused regions from a low DOF image. Focused regions representing the region of interest may not be homogeneous with respect to low-level features such as colour and texture. Therefore, typical segmentation algorithms [15-24] are not suited to this problem. In typical segmentation algorithms, an image is segmented into similar colour and texture regions irrespective of this fact that a region is a part of ROI or background. Figure 1.2 illustrates a number of segmentation results obtained from Normalized Cut method [17].



(a) (b) (c)

Figure 1.2: Illustration of utilising a typical segmentation algorithm (e.g., Normalized Cut Segmentation Technique [17]) over a number of low DOF images. (a)-(c) Segmentation results when using two, three, and five partitions, respectively. The red lines indicate the boundary of partitions. The segmentation process takes more than 20 seconds (on average over the four low DOF images) on a Core2 Due 2.66GHz Intel processor and 2.00 GB of RAM.

There are two main challenges in automatic segmentation of interest regions in a low DOF image that should be addressed. One challenging aspect of this work is dependency on high-frequency contents [25]. Most existing techniques extract focused regions by exploiting high-frequency components. It has been shown that concentrating on high-frequency contents alone often results in errors in both focused and defocused regions. Therefore, the extraction of interest regions should be supported by some supplementary methods or other cues. Another challenge in this type of extraction which has not been adequately addressed is processing time and computational complexity. Proposing a time-efficient approach to extract objects from a low DOF image or video frame is very demanding in a variety of practical multimedia applications. This is especially important since low DOF images are most frequently used within Web images and consequently a real-time processing is required.

1.3 Thesis Aim and Objectives

The original aim of this work is to investigate and develop an efficient meaningful region extraction approach for images with low DOF to support a new region-based functionality in multimedia applications. The project aim includes identifying objects in the image foreground and removing meaningless blurred regions in the background in a way that overcomes the weaknesses of the current low DOF segmentation techniques. The effort toward achieving the thesis objectives includes the following aspects:

- Initial classification of focused regions at the level of block size using a reliable clustering approach

- Develop a novel clustering method that overcome the weaknesses of dependency on initialisation process in clustering methods
- Constructing a binary region saliency map (RSM) from the initial clustering result
- Post-processing step to obtain the shape and boundary of interest regions using morphological operations
- Incorporating the graph cut optimisation technique and colour information into our initial classification approach

1.4 Thesis Contributions

The original contributions presented in this thesis are summarised as follows:

- A novel algorithm was developed and implemented for block-wise low DOF image segmentation. The proposed algorithm can effectively identify ROI and background blocks in a grayscale low DOF image. The proposed algorithm comprises a two-level based ensemble expectation-maximization (EM) clustering technique.
- To augment the visibility of low intensity variations of the regions and boundaries, a novel methodology was developed and implemented. The proposed methodology includes optimising a threshold in difference of Gaussian (DOG) image and using morphological operations.
- The last contribution of this thesis is to develop a colour-based automatic segmentation by incorporating the graph cut optimisation technique into our block-wise EM clustering algorithm.

1.5 Outline of Thesis

The thesis presents the work carried out by the author in attempting to achieve the aim and objectives outlined earlier in Section 1.2. The structure and content of the thesis is described in the following on a chapter-by-chapter basis.

Chapter 2 introduces the DOF technique in photography and its relationship with image segmentation. We also present a number of well known techniques used in this context including an image blurring model, wavelet transform, and also *k*-means clustering algorithm. In addition, the literature review of low DOF image segmentation techniques is provided.

Chapter 3 proposes an ensemble clustering approach which is suitable for efficient block-based low DOF image segmentation. The proposed approach utilises a mixture of Gaussian model, EM algorithm, and an ensemble fusion decision approach. The outcome of this approach is a reliable partition (i.e., clustering result) conforming image objects (i.e., interest regions) at the level of block size.

Chapter 4 aims to extract interest regions at the level of pixels in two different approaches. In the first approach, a DOG method and a threshold optimising technique are firstly employed. Then, to identify the underlying region shape and the boundary of an object, a set of morphological transformations is utilised. In the second approach, a colour-based graph cut modelling is utilised. Visual segmentation results for both approaches are also provided.

Chapter 5 analyses the performance of the proposed approach by adopting a segmentation performance criterion and using two main datasets that include

more than 250 low DOF images. It presents the segmentation performance of the proposed approach in comparison with existing state-of-the-art approaches. To demonstrate the generalisation ability of the proposed approach, we also tested it using a number of images over a specified range of resolutions. Moreover, more than 100 images are chosen to assess the effect of changing the threshold on the performance of the approach.

Chapter 6 discusses and concludes the overall results and contributions. The chapter ends with some pointers and comments to the future work derived from the thesis.

Chapter 2

2. THEORETICAL BACKGROUND AND LITERATURE REVIEW

2.1 Introduction

The aim of this chapter is to review the low DOF technique in photography and its relationship with image segmentation. Low DOF is an important technique used to highlight a certain object of an image. With low DOF, only ROI is in sharp focus, whereas background objects are typically blurred to out-of-focus. In theoretical background, a number of high-frequency based techniques such as DOG and Wavelet, which have the potential to achieve low computational complexity, are presented. *K*-means clustering which is one of the most popular and well-known clustering algorithms is also discussed. Moreover, the literature review representing a number of strong

techniques used in the context of low DOF image segmentation is presented and discussed.

2.2 Depth of Field and Photography

This section describes the characteristics of DOF in photography. Figure 2.1 illustrates a typical imaging system (i.e., single lens camera) containing an image plane and the depth information within a 2D photograph. In a typical imaging system, DOF is defined as the range of distances, behind and in front of the object, from a typical camera lens where the object appears reasonably or fairly sharp [1]. Most professional photographers aim to make use of a wider aperture, longer focal length, or closer camera-to-object distance to put the region of interest in the zone of sharpness (this is why we call this high-level information). In other words, the object points in only one plane (ideal object plane) in front of the camera, which are at the same level of camera-to-object distance, are completely focused (i.e., sharp focused). However, moving farther away from this plane causes blurred discs, i.e., circle of confusion (see Figure 2.1). Accordingly, a reasonable degree of sharpness for all objects within the depth of field zone is obtained.

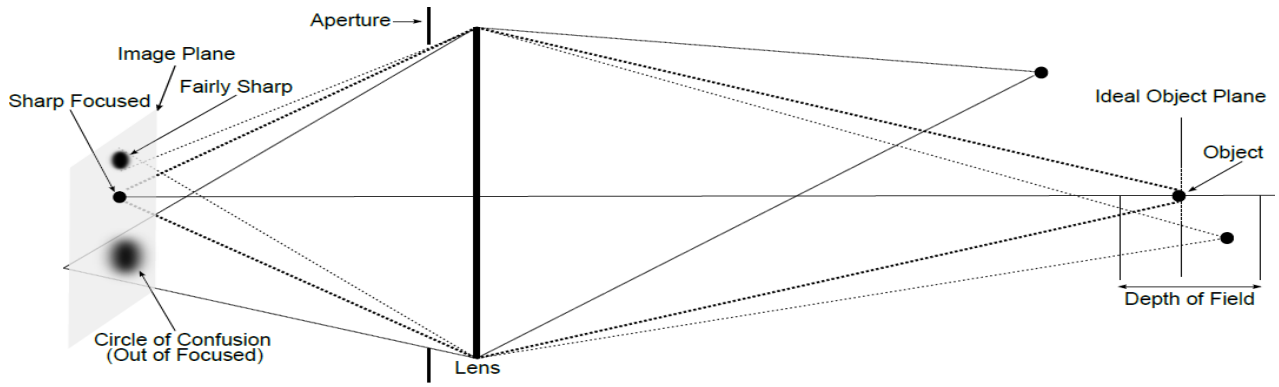


Figure 2.1: Illustration of depth of field in a typical imaging system at maximum aperture.

It is worth mentioning that transition from sharp focused to out of focused regions is usually gradual, which may not even be perceived by the human eye. This gradual transition representing various degrees of sharpness may lead to smooth regions in an image.

Figure 2.2: Three examples of typical low DOF images in grayscale format.

2.3 Image Feature Extraction and Segmentation

Extracting image features/data (such as texture, colour, shape, and salient points) is regarded as a crucial pre-processing step in all content-based image analysis tasks such as object/concept detection, similarity estimation, image indexing, and CBIR systems [7, 9]. In the simplest definition, a feature describes a certain visual property of an image, either for the whole image pixels in a global manner or locally for a small set of pixels.

In a global feature extraction technique [9], a single feature for the entire image, as a global feature, represents the significant characteristics of individual objects in an image. In local extraction, an image is often split in local regions (e.g., blocks) [8] and then features are individually computed for every region. This can be efficiently done by dividing an image into small and non-overlapping blocks. It has been proven that image feature extraction at the level of local region is more promising compare with a global feature extraction in image representation and characterising [9, 26-29].

A procedure subsequently needs to be employed to mathematically formulate the obtained visual features into vectors or distributions so called "image signature" or "visual descriptors". Figure 2.3 depicts an overview of image features, signature types, and mathematical formulation [9].

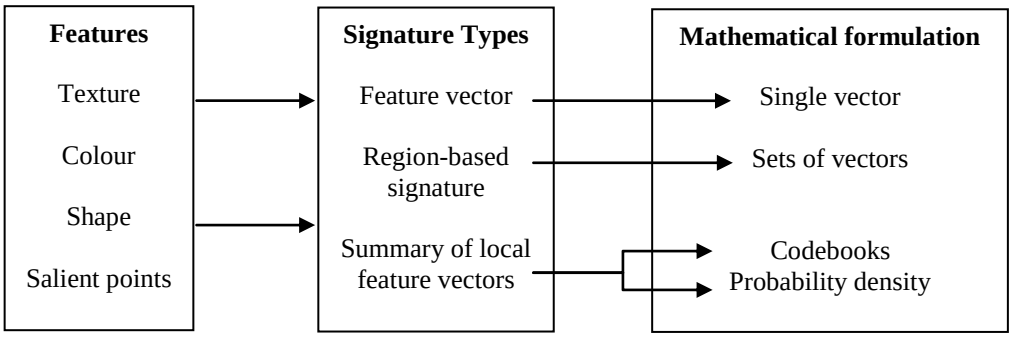


Figure 2.3: An overview of image features, signature types and mathematical formulation [9].

The main goal of image segmentation is to extract relevant/coherent regions, which meaningfully correspond to different objects in an image. Segmentation by object (or relevant regions) is widely regarded as a difficult problem and also an active research area in the image processing community.

Segmentation process plays an important role in many content-based image/video applications such as image indexing (or image annotation), image archiving, surveillance, traffic monitoring, video conferencing, and focus tracking in video frames.

Extracting meaningful and relevant regions is a major step towards image understanding and still remains an open problem [9]. Many segmentation algorithms have been developed based on either focusing on similarity of low-level features [18-25] or incorporating high-level knowledge into the feature proximity process [3-4, 13, 25, 30-48]. This knowledge which assists the reliable segmentation process can be viewed in several forms: task-driven knowledge [30], predefined labels for interactive segmentation [31-35], and low DOF information [3-4, 13, 25, 36-48]. In a typical low DOF image, only the ROI areas are in sharp focus and the remaining areas are blurred and defocused. Accordingly, the ROI areas have more high-frequency components than the defocused areas [25]. This property may be the only reliable clue that can be used to automatically extract the interest regions of an image.

2.4 Image Blurring Model and Difference of Gaussian Function

Blurring effect in an image can be described by a 2D Gaussian function called convolution kernel or point-spread function (PSF) [49]

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{\left(\frac{x^2+y^2}{2\sigma^2}\right)} \tag{2.1}$$

where σ is a filter scale (i.e., standard deviation or scale) which controls the amount of defocusing in a region. An image with defocused regions denoted by $I_d(x, y)$ at a pixel (x, y) can be represented as the linear convolution of an

input image including sharp focused regions denoted by $I(x, y)$ and a Gaussian function $G(x, y, \sigma)$

$$I_d = G(x, y, \sigma) * I(x, y) \tag{2.2}$$

where $*$ is the convolution operation in (x, y) and I_d is also considered as a Gaussian image or the scale-space of the input image $I(x, y)$ [50-51]. It has been shown that by subtracting two Gaussians with two nearby scales separated by a constant multiplicative factor k, a close approximation of Laplacian of Gaussian can be constructed [52]. Accordingly, a DOG image is provided by

$$\begin{aligned} D(x, y, \sigma) &= \big(G(x, y, k\sigma) - G(x, y, \sigma)\big) * I_f(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned} \tag{2.3}$$

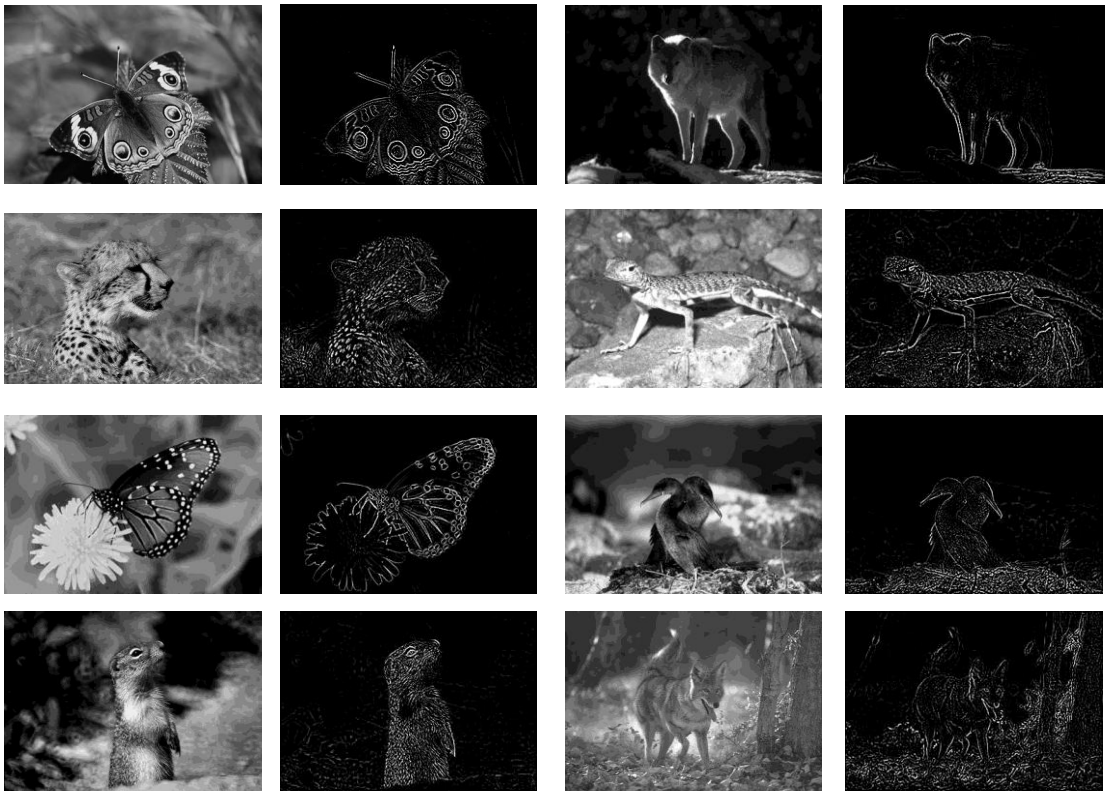
The DOG image in (2.3) can be efficiently constructed by a simple image subtraction. This image represents the various intensity changes of the input image. Figure 2.4 illustrates an example of a grayscale component of a low DOF image, corresponding DOG image, and also its binary image using Otsu's method [53], which is a strong global thresholding method [54]. The low DOF image illustrated in Figure 2.4 includes uniform background and also a sharply focused object. For this image, intensity (luminance) variation is a distinguishing feature that can be utilised to cluster (or segment) the object from its background. The parameters k and σ in (2.3) are chosen to be 0.8 and 0.5 respectively for all these experiments [55]. Consequently, if the sharply focused regions contain sufficient high-frequency components, it would be possible to

distinguish the focused regions from the defocused ones by evaluating or comparing the amount of high-frequency contents.

(a) (b) (c)

Figure 2.4: An example of the grayscale component of a low DOF image. (a) Original low DOF image. (b) Corresponding DOG image. (c) Its binary image obtained from Otsu's method [53].

Practical problems occur, however, when the observed image is subject to busy-texture areas (i.e., noise) and when the object and background assume some broad range of intensity values. Figure 2.5 shows a set of low DOF images including non-uniform/complex background (i.e., busy-texture regions) and smooth object boundary selected from the Corel dataset [56-57]. The corresponding DOG images are also obtained from (2.3) (Figure 2.5(b), (d)). As evident from the Figure 2.5(b), (d), there may be smooth areas in object regions that generate errors in these regions. In background regions, despite the blurring due to the defocusing, there can be busy-texture regions with high-frequency components. Therefore, considering high-frequency components in a low DOF image alone often results in errors in both focused and defocused regions.



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

Figure 2.5: Illustration of various low DOF images. (a) and (c) Graysacle components of a number of low DOF images selected from the Corel Dataset. (b) and (d) Corresponding DOG images.

2.5 Wavelet Transform in Digital Image Processing

Wavelet transform is a powerful tool for texture analysis and representation [58-60]. A wavelet, i.e., small wave, is defined as an irregular and asymmetric waveform of effectively limited duration that has an average of zero. Wavelet analysis consists of breaking up a signal into shifted and scaled versions of the original/mother wavelet. Figure 2.6 illustrates a selection of common mother wavelets used in practical applications. The wavelet transform of a continuous signal, $f(t)$, is defined as [61]

$$F_{CWT}(\tau, s) = \frac{1}{\sqrt{|s|}} \int_{-\infty}^{+\infty} f(t)\psi\left(\frac{t-\tau}{s}\right) dt \text{ where } \tau, s \in \Re\ (s > 0) \qquad (2.4)$$

where the original wavelet is denoted by $\psi(t)$ and the factor $1/\sqrt{|s|}$ is used to conserve the norm. The parameters τ and s denote the location of the wavelet in time and scale, respectively. The elements in $F_{CWT}(\tau, s)$ are called wavelet coefficients, each wavelet coefficient is associated to a scale (frequency) and a point in the time domain. The global information of a signal can be characterised by a large scale corresponding to a low frequency. Small scales correspond to high frequency which can provide the details of a signal [58].

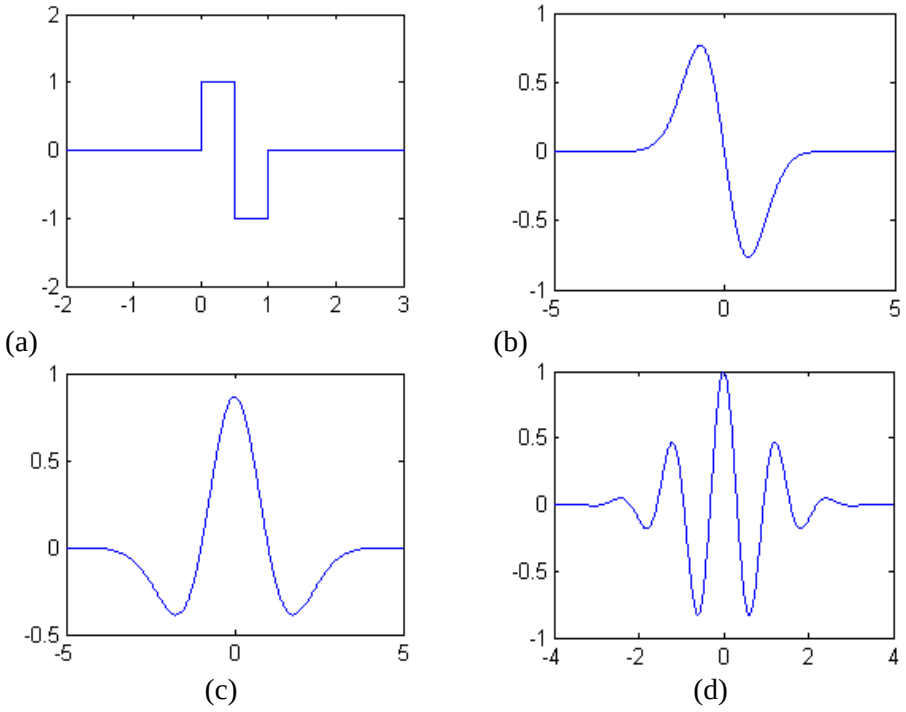


Figure 2.6: Four mother wavelets. (a) Harr. (b) Gaussian wave (first derivative of a Gaussian). (c) Mexican hat (second derivative of a Gaussian). (d) Morlet (real part)

A fast and practical discrete wavelet decomposition and reconstruction algorithm introduced by [61]. This algorithm is a classical scheme known in the signal processing community as a two-channel subband coder. The

decomposition step generates two sets of coefficients called ‘approximation’ and ‘detail’ vectors by convolving the original signal x with a low-pass filter and a high-pass filter, respectively, followed by dyadic decimation, i.e., removing every odd element of an input sequence, as illustrated in Figure 2.7. Reconstruction of the original signal is accomplished by upsampling, filtering, and summing the individual subbands [58].

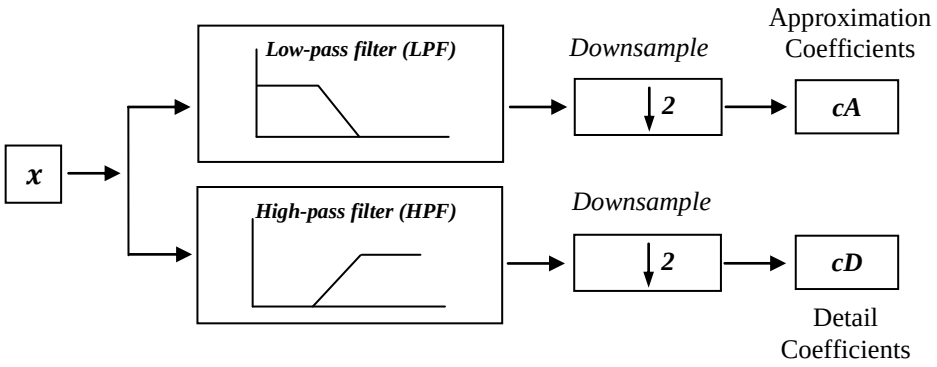


Figure 2.7: One-level decomposition algorithm introduced by [58]. Downsampling process keeps the even indexed elements to reduce the overall number of computations.

Using the same idea, the two-dimensional wavelet coefficients of an image can be achieved by employing separable filters for each dimension (i.e., row and column) along with downsampling process as illustrated in Figure 2.8. The outputs of the decomposition algorithm in Figure 2.8, denoted cA, cD^V, cD^H and cD^D, are called the approximation, vertical detail, horizontal detail, and diagonal detail subbands of the image, respectively.

In Figure 2.9, four-band split of the *butterfly* image using the Haar wavelet [26] and one-level decomposition algorithm has been illustrated. This function has been successfully used in a number of image processing applications due to its low computing requirements [3, 57, 62]. To obtain multi-level wavelet

transform, a successive approximation band being decomposed into four smaller subbands, which can be split again, and so on [58]. Figure 2.10 shows a two-level decomposition wavelet transform for the *butterfly* image. As illustrated in Figure 2.10(c), the high-frequency band information in the diagonal direction (located in the bottom right corner) demonstrates interest regions (i.e., a butterfly and a leaf) without any background noise for the *butterfly* image. This is because there is no any noisy region in the background in the diagonal direction for this image.

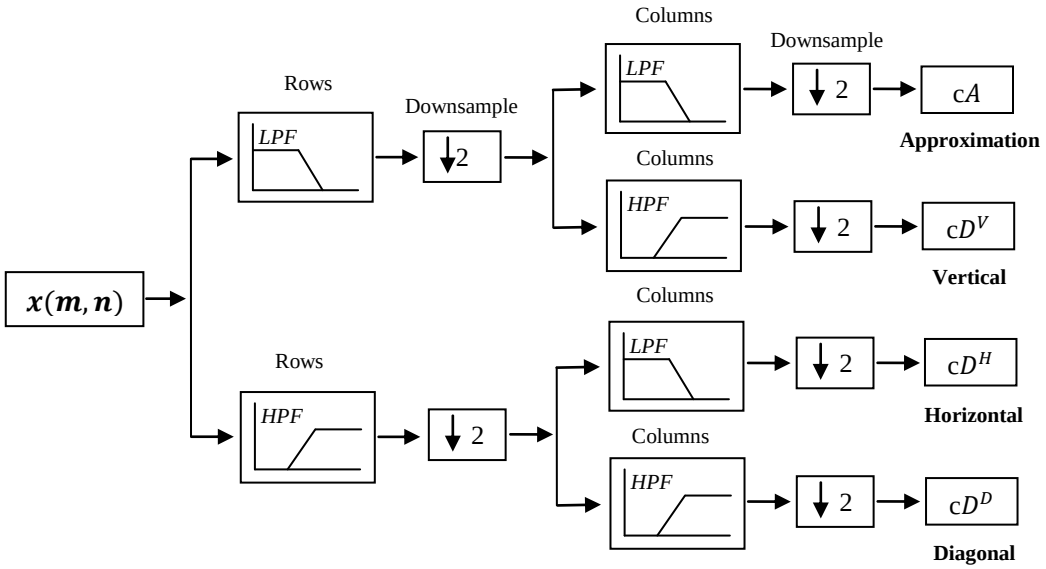
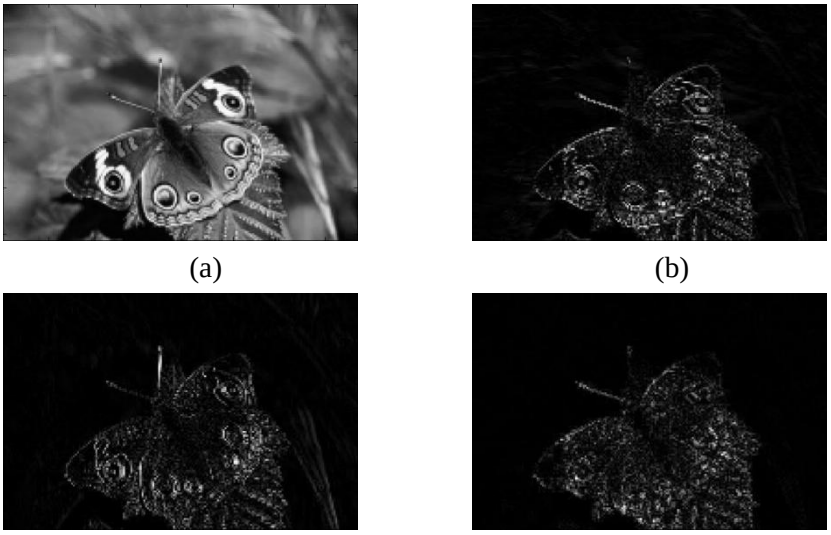
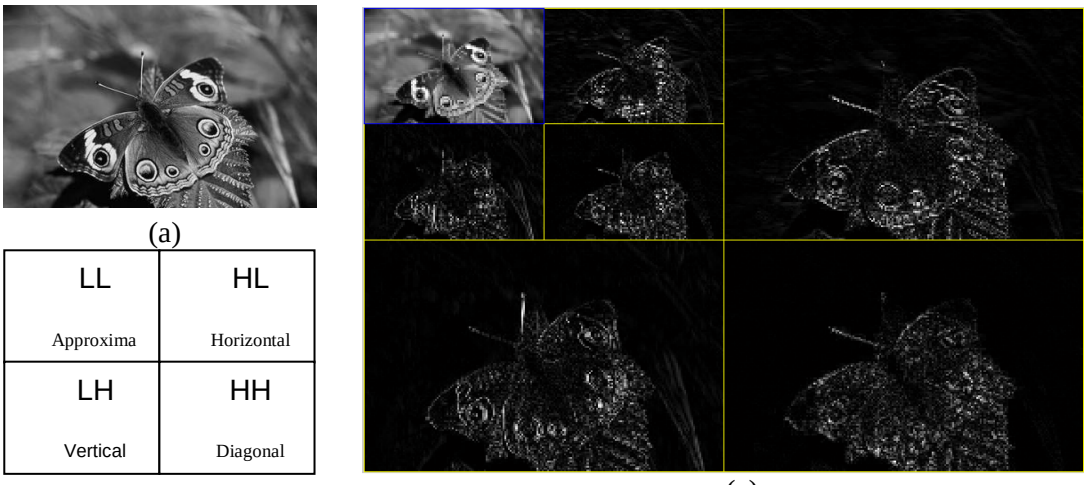


Figure 2.8: Illustration of a single level DWT decomposition algorithm for a given image introduced by [58].



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

Figure 2.9: Four-band split of the butterfly image using the Haar wavelet and decomposition algorithm of Figure 2.8. (a) Approximation information. Detail information in horizontal (b), vertical (c) and diagonal (d) direction.



(a) (b) (c)

Figure 2.10: Wavelet representation on two resolution levels. (a) Original grayscale *butterfly* image. (b) Arrangement of three high-frequency bands LH (vertical), HL (horizontal), and HH (diagonal) of the wavelet coefficients. (c) Two-level decomposition wavelet transform, level separated by borders.

2.6 *K*-means Clustering Algorithm

Data clustering, also called cluster analysis, segmentation analysis or unsupervised classification/learning is a technique of creating groups of objects or clusters, in such a way that objects in one cluster are very similar and objects in different clusters are quite distinct [63]. Many clustering techniques have been and are being developed in the context of image processing and computer vision (e.g., object categorisation or low level segmentation) [64]. In this context, unsupervised clustering techniques have been categorised into three different types [9]: pair-wise distance-based, optimisation of an overall clustering quality measure, and statistical modelling. *K*-means clustering algorithm is an example of optimisation-based techniques that has been used in this thesis. The low computational complexity of this algorithm makes it an attractive candidate for a variety of applications [64].

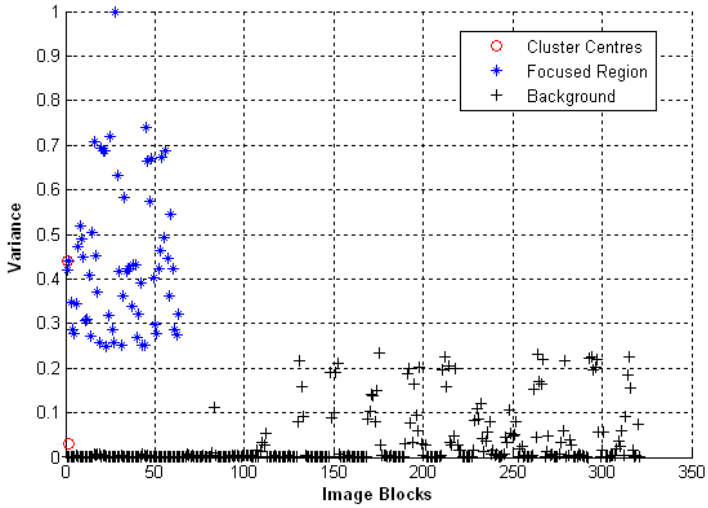
In standard *k*-means clustering algorithm, the number of clusters k is assumed to be fixed. There is an error function (i.e., objective function) which shows how good a clustering solution is. In this algorithm, an initial k clusters is firstly selected. Then the remaining data are allocated to the nearest clusters and repeatedly the membership of the clusters is changing according to the objective function until the function does not change significantly or the membership of the clusters no longer changes [63]. The following demonstrates the procedure of standard *k*-means clustering algorithm.

K-means Clustering Algorithm

Given a dataset D, Number of Cluster k, Dimensions d,
$\{C_i$ is the i-th Cluster$\}$
1: **initialisation phase**: $\{(C_1, C_2, \dots, C_k) =$ Initial Partition of D$\}$
2: **repeat**
3: $d_{ij} =$ Distance between data point i and cluster j
4: $n_i = \arg\min_{1 \le j \le k} d_{ij}$
5: Assign data point i to cluster n_i
6: Re-compute the cluster means of any changed clusters above
7: **until** no further changes of cluster membership occur in a complete iteration
8: Output results

Despite the computational simplicity of this clustering algorithm, it suffers from a major disadvantage which is sensitivity to initialisation process. Different initialisation can lead to different final clustering results, each corresponding to a different local minimum, because this algorithm only converges to local minima. One way to address the local minima of the k-means clustering algorithm is to run this algorithm with various initial partitions and choose the partition with the smallest value of the error [65]. Figure 2.11 illustrates different clustering results obtained from running the *k*-means clustering algorithm on the *butterfly* image. In this experiment, the variance of high-frequency components of wavelet coefficients is used as a feature for the clustering process. The image block size has been also set to 16×16.

(a)



(b)

(c)

Figure 2.11: Illustration of a clustering result using running the *k*-means clustering algorithm on the *butterfly* image. (a) Original grayscale image. (b) Distribution of energy blocks after 12 iterations of the algorithm. Focused and background regions have been illustrated by blue and black colours, respectively. (c) Clustering result at block size level.

(a)

(b)

(c)

Figure 2.12: Illustration of clustering results (block-based) in different runs of the *k*-means clustering algorithm. (a)-(c) Converging after 5, 10, and 11 iterations, respectively.

2.7 Related Research

Several attempts have been made to address the difficulties of low DOF image segmentation [4], [3], [40-44]. In [4], an unsupervised edge-based segmentation approach using the moment-preserving principle has been developed, in which the amount of blur/defocus in an object boundary is evaluated. In this approach, an observed image is firstly converted into a gradient map using the Sobel edge detector [49]. Then, the amount of blur for every edge pixel in an interest region is measured by the proportion of the edge region in a small neighbourhood window using the moment-preserving method. Then, an edge linking procedure including dilation, thinning, line linking, and superimposing is employed to assemble focused edge pixels into closed boundaries. Finally, a filling procedure is used to eliminate all pixels outside of the closed boundaries. Figure 2.13 illustrates the extraction process for a sample image.

As reported by [3], this approach is only suitable for segmenting closed boundary objects. Wang et al. [3] proposed a region/block-based multiscale approach that detects the sharply focused objects in a low DOF image. This method includes two main steps (See Figure 2.14):

1) Initial grouping with a large block size using a block-based *k*-means clustering. 2) A multiresolution refining approach for segmenting results obtained in the previous step.

In the initial grouping step, *k*-means clustering algorithm and the variance of high frequency wavelet coefficients of each block as a feature have been employed. The second step, which uses the average intensity of each block, includes an iterative process to find the similarity between blocks.

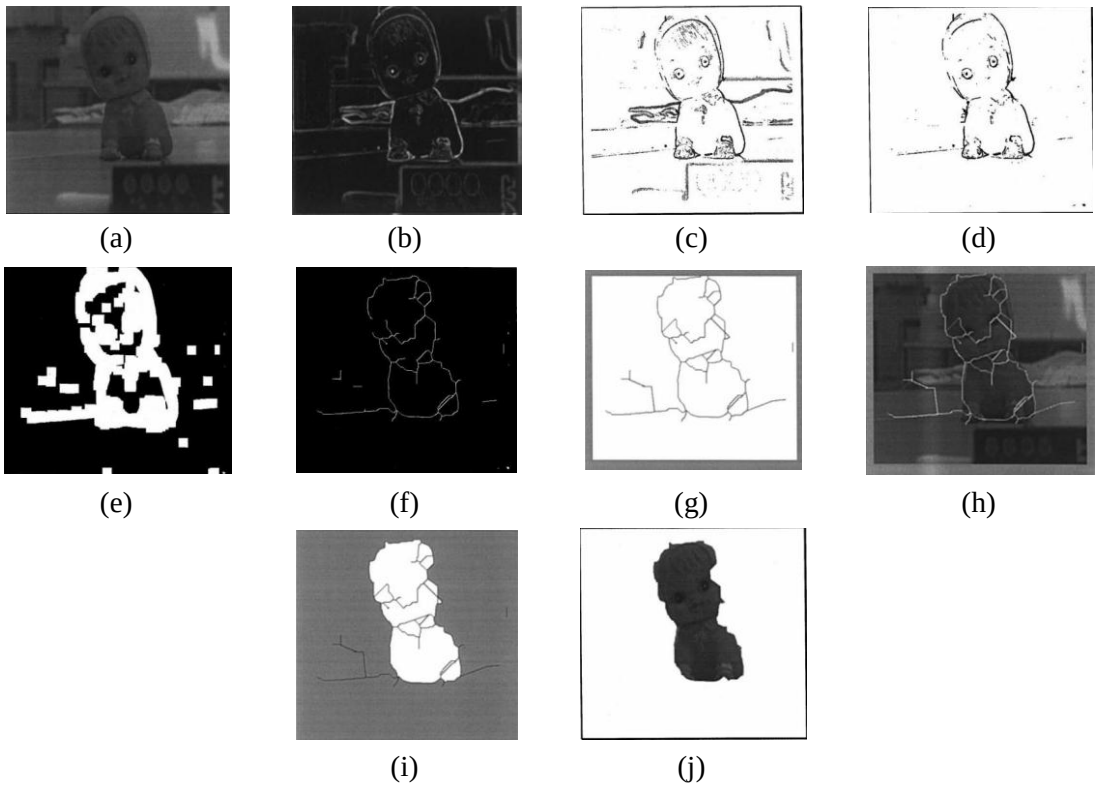


Figure 2.13: The extraction process for a sample image using moment-preserving principle [4]. (a) Original grayscale image. (b) Gradient image using Sobel edge detector. (c) P_e image, P_e is an indicator for the diameter of blur area. (d) Thresholding result of (c). (e) Dilation result. (f) Thinning result. (g) Edge linking. (h) Superimposing (g) on (a). (i) Filing the background. (j) The extracted object.

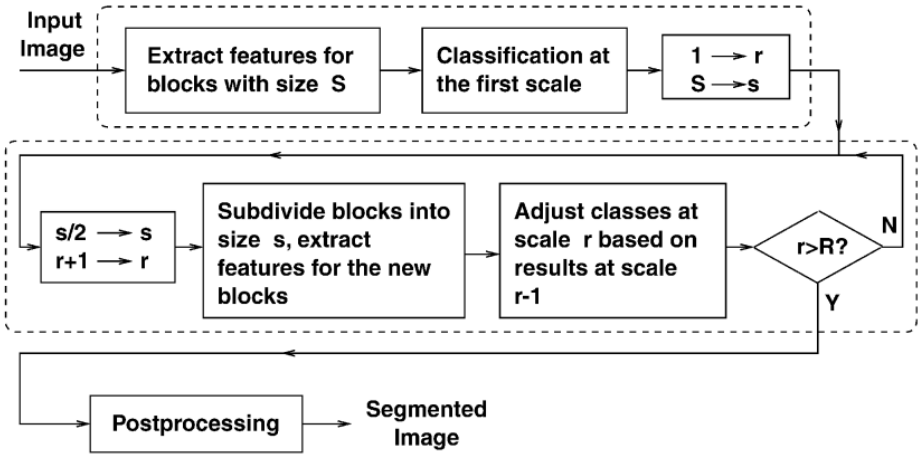


Figure 2.14: The main steps of the unsupervised multiresolution segmentation approach proposed by [3]. The block size S is initialised by 32×32 and is subdivided into child blocks at every increased scale (i.e., $r = r + 1$). The maximum scale is defined as $R = 6$.

Although their experimental results manifested the efficiency of the algorithm, their proposed method has several limitations, which can lead to unreliable segmentation. The most important limitation lies in the fact that the algorithm considers only two classes of regions including out-of-focus objects and sharply focused object of interest. Therefore, if a block of object-of-interest includes partially sharp focused or smooth regions, the algorithm is unsuccessful in extracting the object. Moreover, as the initial grouping process has been carried out by *k*-means algorithm, it may be trapped into a local optimum and consequently an undesirable result may be obtained. Figure 2.15 shows the schematic of segmentation results using k-means clustering algorithm at lowest scale and a multiscale approach. Figure 2.16 and 2.17 illustrate a number of segmentation results using this approach.

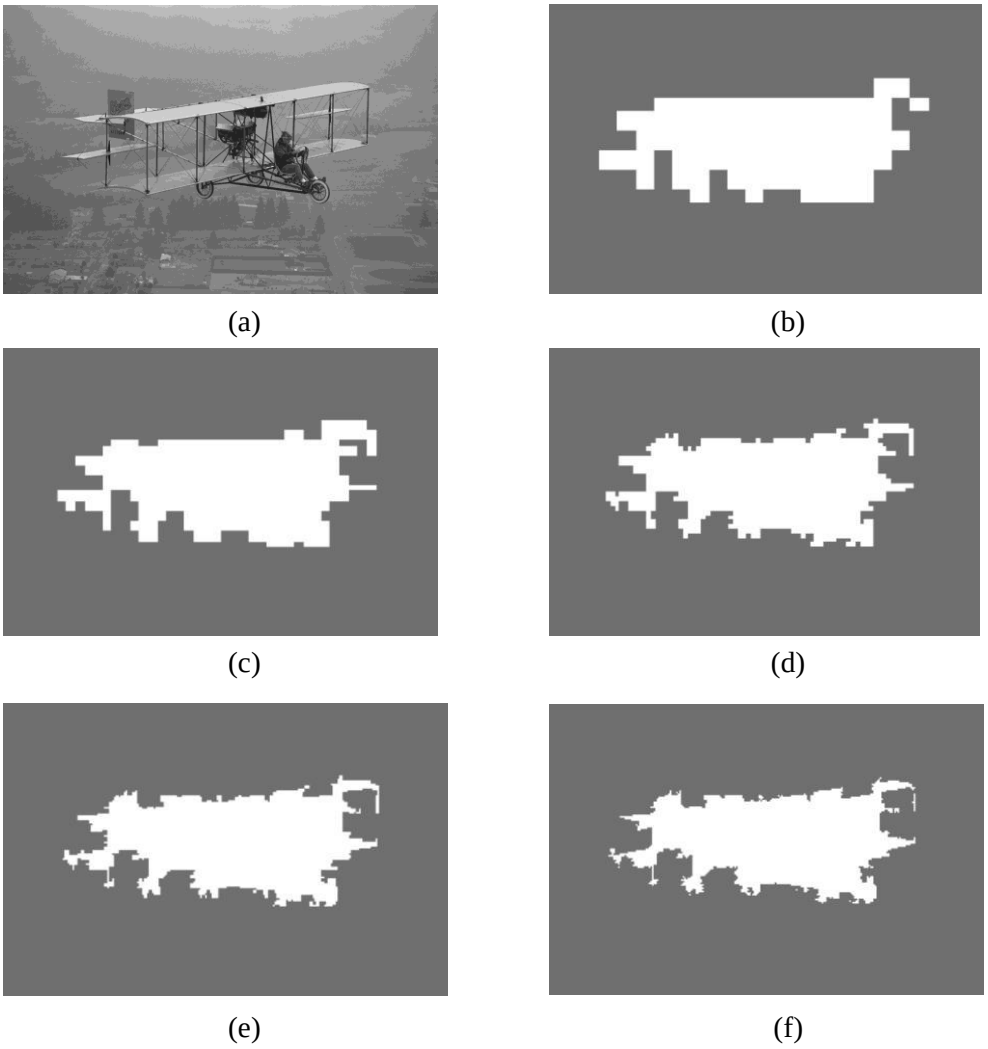


Figure 2.15: The sequence of segmentation results for a sample image using the multiresolution segmentation approach [3]. (a) Original grayscale image. (b) Initial classification at the lowest scale (block size=32) using the k-means clustering algorithm. (c)-(f) a recursive process to adjust the crude classification result using a multiscale approach.

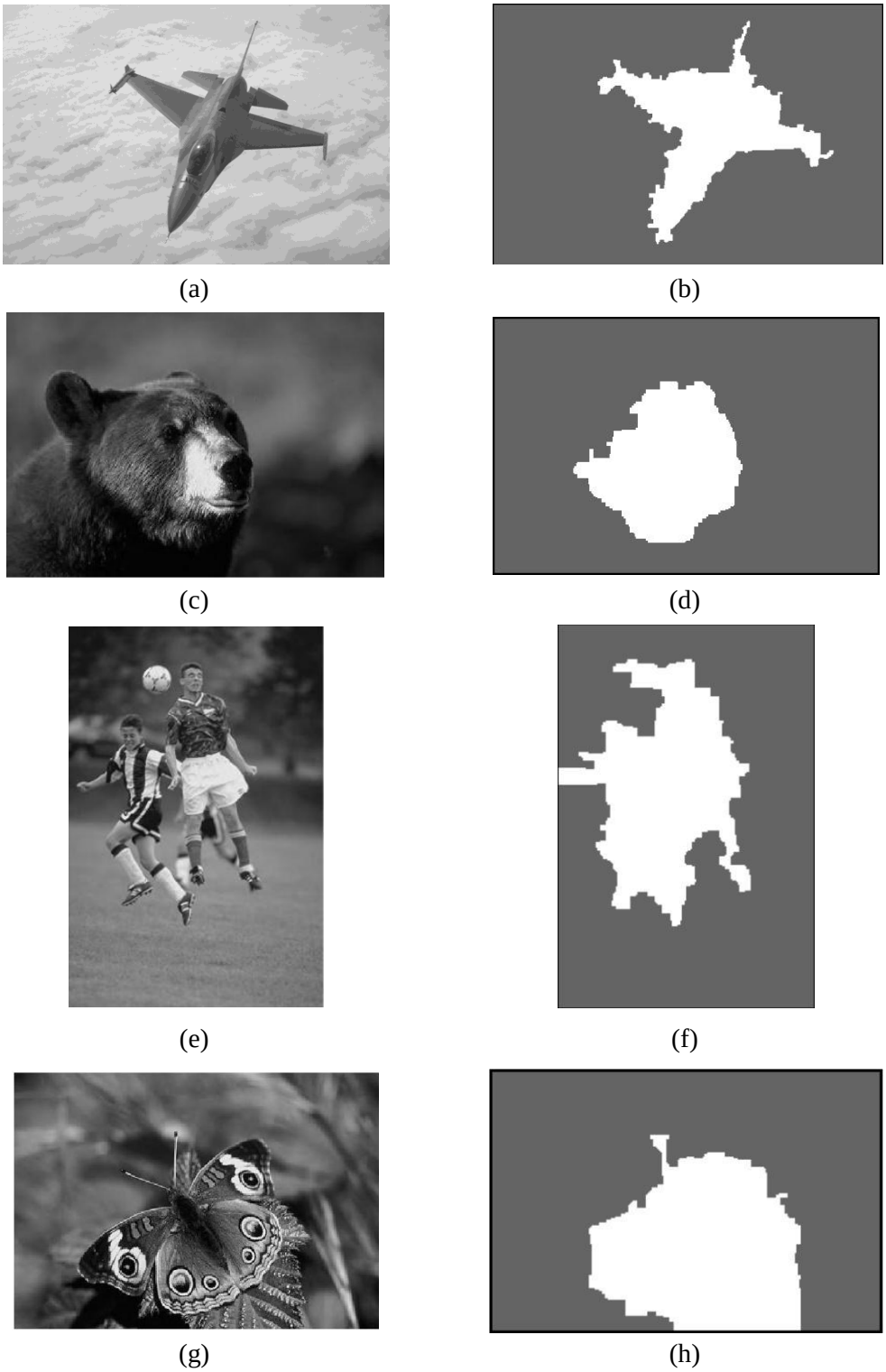
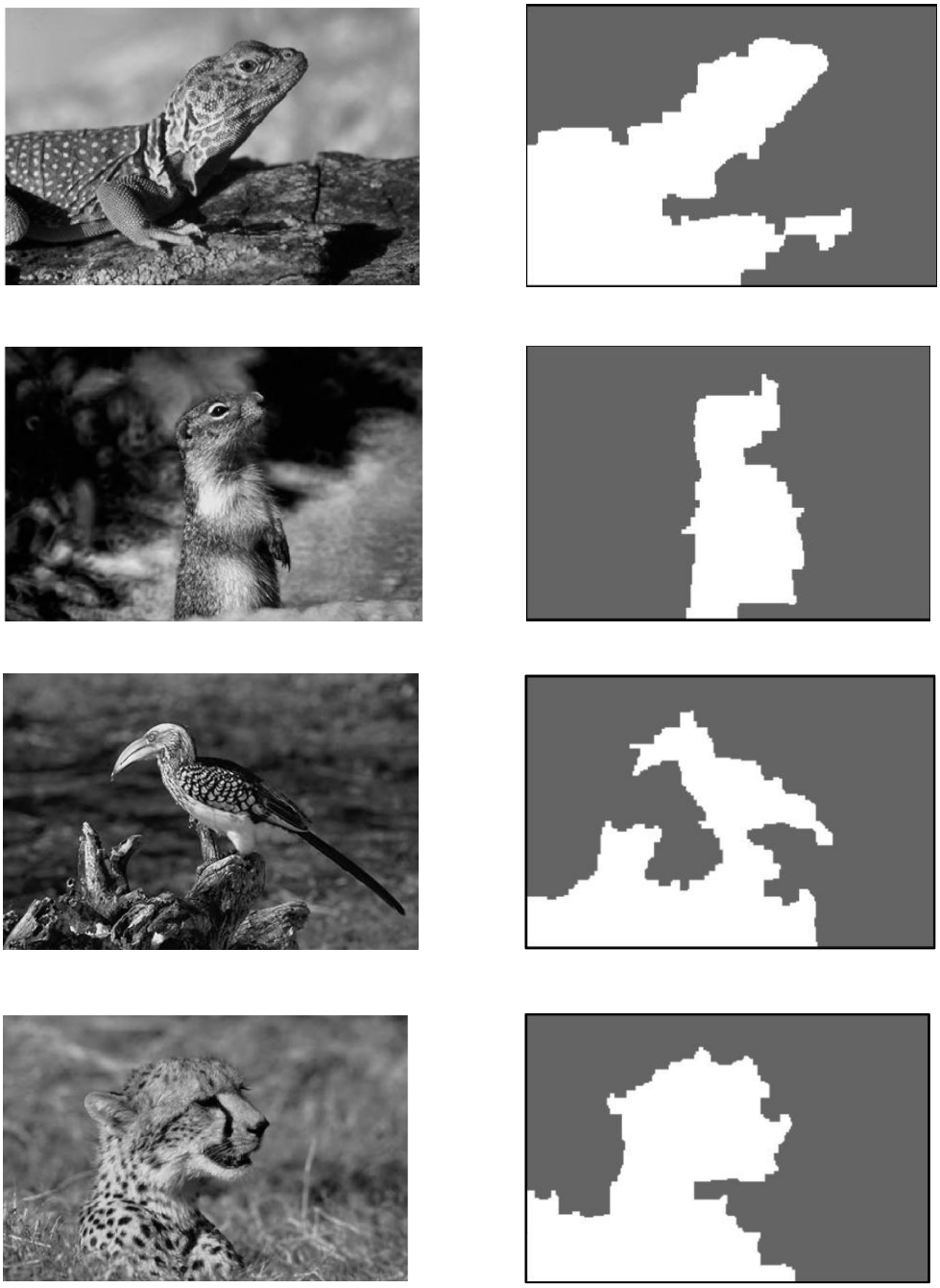


Figure 2.16: Segmentation results for a number of low DOF images obtained from the unsupervised multiresolution segmentation approach [3]. (a),(c),(e), and (g) Original grayscale images. (b),(d),(f), and (h) Corresponding segmentation results.



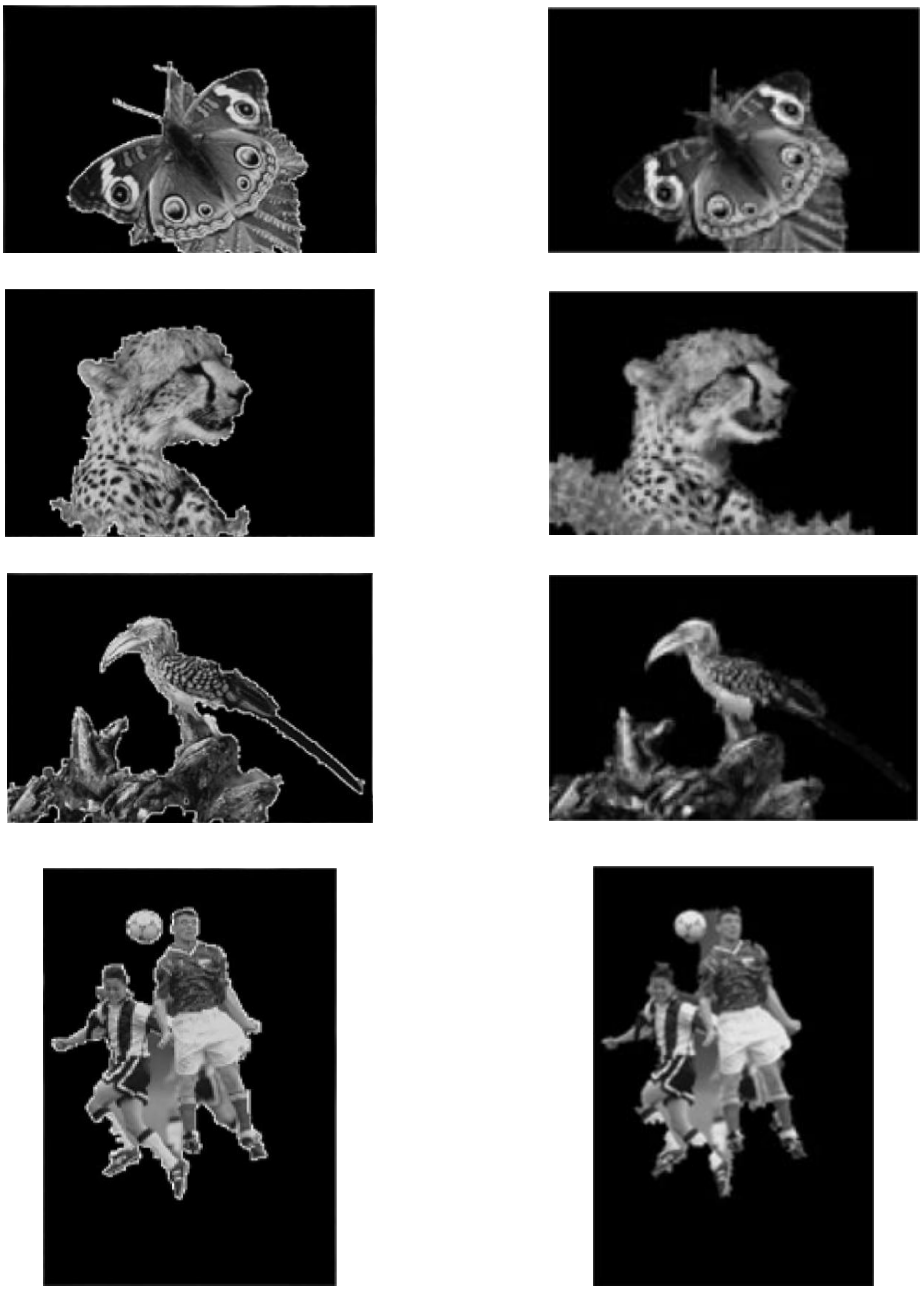
(a) (b)

Figure 2.17: Segmentation results for a number of low DOF images obtained from the unsupervised multiresolution segmentation approach [3]. (a) Original grayscale images. (b) Corresponding segmentation results.

In [40], the author presented a pixel-based approach in which three steps have been employed to partition all image pixels into focused object-of-interest and defocused background. Firstly, high-frequency components are extracted using computing the forth-order moment of all image pixels (i.e., higher order statistics map). Then, a simplification process using morphological filter is used to remove the errors originated from smooth focused and defocused regions. Finally, to extract the object of interest, region merging and thresholding processes are applied to the simplified image. Despite more accurate results compared with the previous methods such as [4] and [3], the algorithm needs to have a sufficiently defocused background and consequently is not suitable for segmenting an image with busy-texture (i.e., noisy) regions in its background. Moreover, as the first stage of this approach is based on computing the higher order statistics map of all image pixels, extracting focused regions requires high computation complexity.

The pixel-wise segmentation algorithm in [41] aims to extract the focused objects in low DOF video images based on a matting equation, in which three stages have been considered. To create a saliency map using a re-blurring model is the first stage of the algorithm. Then, bilateral and morphological filters are used to smooth and merge the regions of the map. The final stage attempts to extract the boundaries of the focused object using an adaptive error control matting scheme. As pointed out by the authors, this method presents a better performance than the method presented in [40]. However, the proposed segmentation method, which is based on a matting model, suffers from high computational complexity, leaving the low-complex extraction of focused areas

mostly unaddressed. Figure 2.18 (a)-(b) illustrate the visual segmentation results of [40] and [41], respectively.



(a) (b)

Figure 2.18: Visual comparison of segmentation results. (a) Results obtained from [40]. (b) Results obtained from [41].

In [42], a supervised learning approach has been presented to extract attention objects from low DOF images by employing three types of visual features including texture, colour, and geometrical property. In their approach, they firstly used a supervised learning algorithm to train a cascade of three classifiers. Subsequently, the classifiers are employed to identify defocused regions in three phases. Finally, region grouping and pixel-level segmentation procedures are carried out to improve the accuracy of results. Their experimental results based on 89 training images and 117 test images demonstrated a good accuracy with 4.635 seconds on a Quad CPU 2.66 GHz as the average computation time for extracting focused objects in low DOF images. Figure 2.19 and 2.20 illustrate the framework of the approach proposed and a number of segmentation results obtained from [42].

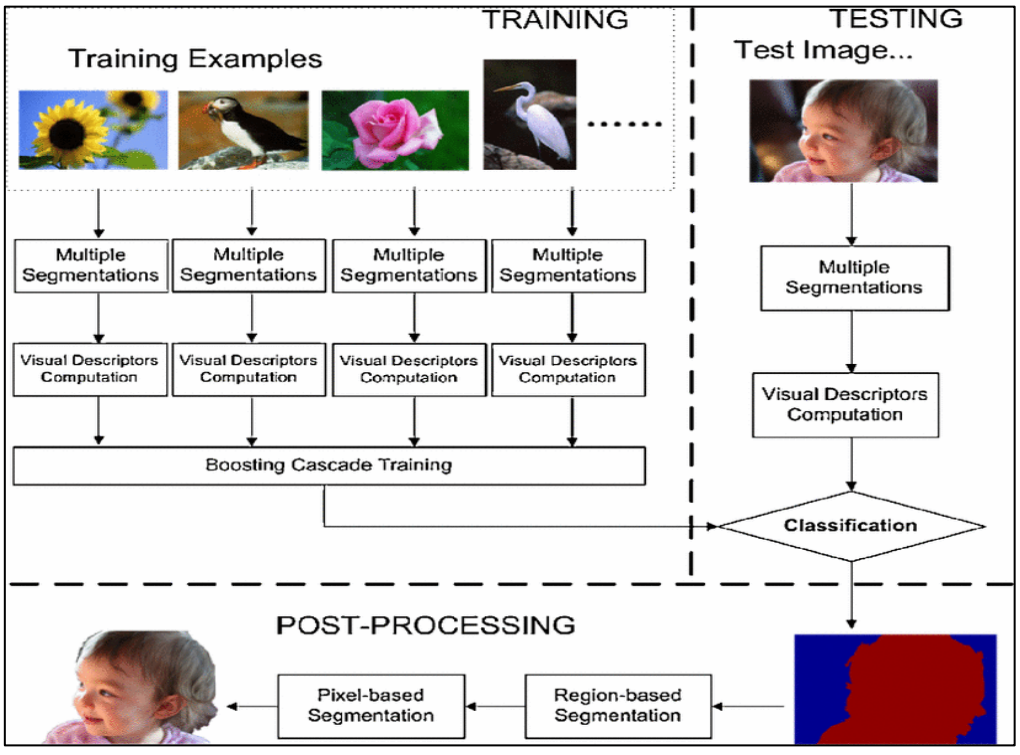


Figure 2.19: Illustration of supervised framework proposed by [42].

(a) (b)

Figure 2.20: Illustration of segmentation results obtained from the approach [42]. (a) Manually segmented images (i.e., ground truth masks). (b) Segmentation results using the supervised learning approach.

The algorithm proposed in [43] aims to segment image pixels into region of interest and background in three main stages. In the first stage, sharp pixels are identified by using a Gaussian kernel and a clustering approach. The second stage of the algorithm generates a binary mask by connecting isolated clusters and also using morphological operations. Finally, the obtained mask is refined by using a colour segmentation and region scoring technique. Experiments conducted on a dataset of 65 low DOF images show the superior robustness of the algorithm compared with [41]. However, this algorithm is computationally expensive and requires high computational time.

Recently, an unsupervised method [44] based on an amplitude decomposition model, thresholding process, and graph-cut technique has been presented to extract focused objects from low DOF images. Their experimental results show the method is comparable to the state-of-the-art methods. However, this method may fail when the background has a similar colour distribution with the focused objects. This method takes an average of about 7 seconds to segment a low DOF image of the size 400×300, which is much faster than [41] and comparable to [40].

2.8 Image Dataset

The selection of an image dataset covering a wide range of busy-texture, smooth regions, and complex background is still very controversial issue in low DOF image segmentation community. A considerable amount of literature has been published based on small categories of image datasets. Among the image datasets, Caltech [66-67], PASCAL VOC [68], Berkeley dataset [69], and Corel database [56-57, 70] have large diversities and classes which have been widely employed in object segmentation recognition, classification, image retrieval, and annotation purpose [71]. The Corel dataset collection includes about 60,000 pre-classified images from various concepts collected by Wang et al. [72]. In each category of the collection, 100 images have been used. The Corel images have been extensively employed as a benchmark by researchers in the field of image segmentation, image retrieval and annotation. In this thesis, a number of low DOF images with complex background from Corel dataset have been selected to evaluate the proposed approach. Moreover, more than 100 low DOF Web images along with their manually segmented images which have been provided by [42] have been used to provide an empirical comparison with the state-of-the-art approaches.

2.9 Summary

In this Chapter, the characteristic of the low DOF technique in photography has been presented. The low DOF is an important technique which assists a reliable segmentation task by incorporating some high-level knowledge into a feature proximity process. Moreover, DOG and wavelet functions as high-frequency based techniques for texture analysis and representation have been discussed. Despite all the research and development in the context of low DOF image segmentation, extracting automatically focused regions from low DOF images in an efficient and effective way remains unsolved. In the coming Chapter, an efficient and effective solution based on ensemble clustering will be presented.

Chapter 3

3. ENSEMBLE CLUSTERING APPROACH

3.1 Introduction

In this chapter, a novel ensemble clustering approach is developed and implemented to extract important objects from a low DOF image at the level of block size. The main focus of this Chapter is to address the common problem of local optima experienced in many models. To achieve this, we model the joint distribution of image features (i.e., contrast and energy) in a certain level with a mixture of Gaussian. EM algorithm is also utilised to find the parameters of the mixture model and iteratively create different local optimum solutions based on

various initial configurations in each level of resolution. Finally, a fusion decision approach is developed to combine different solutions; this results in a final clustering result for a low DOF image at the level of block size. The proposed approach is tested on over 250 low DOF images from two main datasets.

3.2 Overview of the Proposed Approach

In low DOF images, focused regions typically represent important object(s) in a foreground. Generally, these areas include focused object(s) of interest and possibly some focused regions in the background which are informative and mostly relevant to the focused object in the foreground (as illustrated in Figure 3.1). For convenience, we call the focused regions the **ROI** and the defocused areas the **background** in this thesis. In this Chapter, local regions (i.e., blocks) in an image are classified into three constituent classes which can be used to differentiate the ROI from the background at the level of block size. This grouping, which is a crude segmentation process, is achieved by a two-level based ensemble EM clustering technique. We only use the grayscale component of the original colour image for extracting texture features. Figure 3.2 illustrates the main components of the proposed segmentation approach.

(a) (b)

Figure 3.1: (a) Illustration of a grayscal image (namely *butterfly*) and (b) ROI and background at the level of block size.

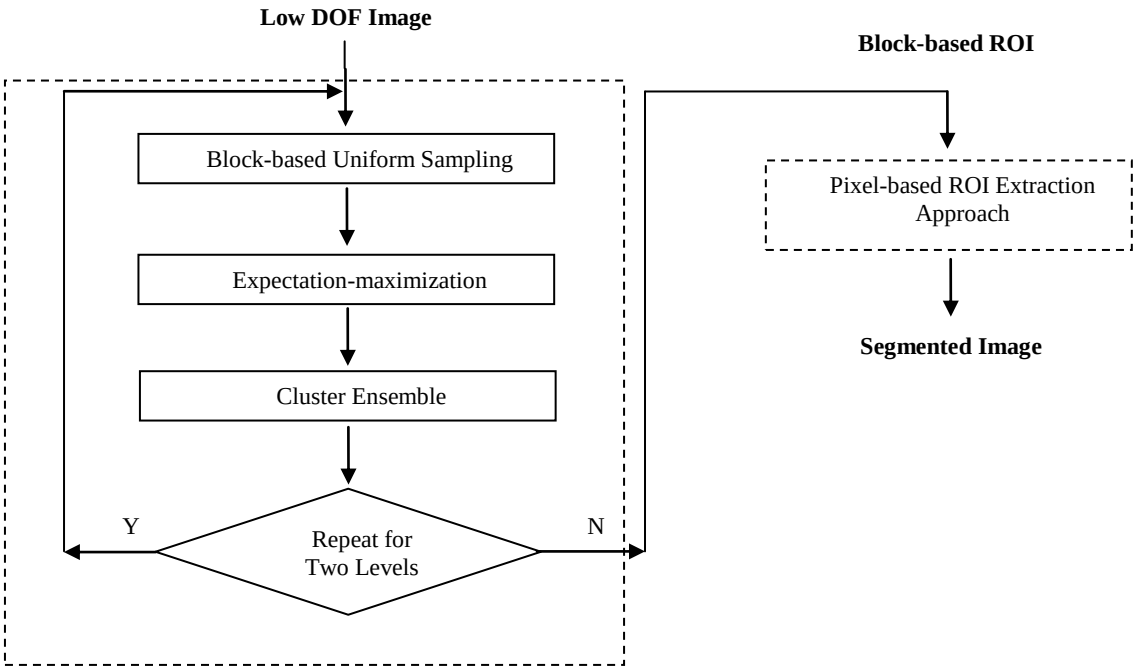
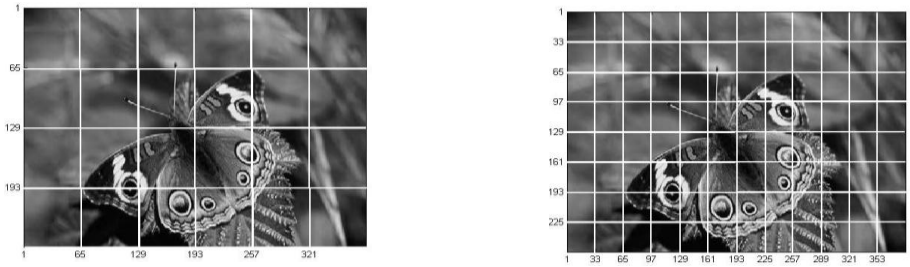


Figure 3.2: The main components of the proposed segmentation approach. Block-based ensemble EM clustering technique at two levels (left) and pixel-based ROI extraction approach (right).

3.3 Region Sampling and Characterising

Suppose an image of $M_I \times N_I$ pixels is represented by a set of uniform and nonoverlapping blocks. In our approach, the initial block size, i.e., $S_B = s \times s$, covers approximately 17% of the maximum number of pixels in rows and columns of a grayscale image. For example, the block size for an image with the size of either 384×256 or 256×384 pixels is chosen to be 64×64. As we need to capture the texture details of an image at the level of object, the feasible large block size is selected. We also choose the next consecutive level of block size of a selected image by dividing each block of the first level into four blocks. Figure 3.3 illustrates two consecutive levels of block sizes for the butterfly image selected from the Corel dataset.



(a) (b)

Figure 3.3: Illustration of the two consecutive levels of block sizes for the butterfly image of size 384×256. (a) and (b) First and second levels including the block sizes of 64×64 and 32×32, respectively.

To extract the details of texture in each block, we make use of the conventional contrast definition, which is demonstrated in [45, 55], and also wavelet coefficients as block descriptors [56, 73]. This selection allows us to have a good compromise between effectiveness and time efficiency. The dynamic range of gray levels in a block representing the local contrast can be

approximated by

$$x_c = \frac{I_{max} - I_{min}}{I_{max} + I_{min}} \tag{3.1}$$

where I_{min} and I_{max} denote the minimum and maximum intensity values of all pixels in a block. To extract the energy of a block, discrete wavelet transform (DWT), which is a multiscale frequency decomposition technique, is employed [60]. The high-frequency bands of wavelet coefficients denoted by $HFB = \{HL, LH, HH\}$ are able to capture the intensity variations of a block in horizontal, vertical, and diagonal directions. The lowest computational cost wavelet transform, the Haar transform [62], with one level of resolution is used to decompose a block containing M_L and N_L pixels into four frequency bands with size $\frac{M_L}{2} \times \frac{N_L}{2}$. The square root of the second order moment of wavelet coefficients in each *HFB* is defined as [57]:

$$f_k = (\frac{1}{4} \sum_{i=0}^{\frac{M_L}{2}-1} \sum_{j=0}^{\frac{N_L}{2}-1} W_{k,i,j}^2)^{\frac{1}{2}} \tag{3.2}$$

where $k \in HFB$ and W_k denotes the matrix of wavelet coefficients of the block in the *HFB* set. The texture descriptor representing the minimum energy of the block, i.e., local energy, can be calculated by

$$x_e = \min f_{\mathrm{k}} \quad , k \in HFB \tag{3.3}$$

where f_k is the energy of the high-frequency bands. A two dimensional contrast-energy feature vector, i.e., $x = (x_c, x_e)$, is then used in a block-wise clustering approach to capture interest regions in an image.

3.4 Region Definition and Clustering

All blocks of a low DOF image are divided into three region classes: out-of-focus (defocused/background), sharp focused (sharp), and uncertain. The out-of-focus regions usually contain uniform intensity or small variations in gray-level values and are generally related to the areas which are often not the main interest of a photographer. In contrast, sharp focused regions characterized by distinctive contrast and energy features represent the significant aspects of the image (i.e., image object). Regions including partially sharp-focused and smooth properties are referred to as uncertain because these sorts of regions suffer from lack of adequate distinctive features in unsupervised learning. In our experiments, the term class can be interchangeable with the term cluster.

In order to automatically cluster the blocks into three classes, we model the joint distribution of contrast and energy features of image blocks, i.e., observed data vector, with a mixture of Gaussian, which has been widely used and acknowledged in the context of statistical unsupervised learning [74-75]. The EM algorithm [74], which can be considered as a soft version of standard *k*-means clustering algorithm [76-78], is also used to find the parameters of the mixture model. In standard *k*-means clustering algorithm [63], each observed data vector is assigned to a group with a probability selected from {0, 1}. In contrast, for the EM algorithm, the assignment of data vector to a component (i.e., cluster/group) is based on a probability selected from [0, 1]. In our approach, the EM algorithm aims to estimate the maximum likelihood parameters of a three-component bivariate mixture. Let $X = \{x^i\}_{i=1}^{R}$ be the observed data, where R is the number of blocks and x^i is a two dimensional

contrast-energy feature vector of a block. Let also $\varphi(x|\theta_m)$ be a Gaussian component parameterized by a mean and covariance of a normal distribution, i.e., $\theta_m = \{\boldsymbol{\mu}_m, \boldsymbol{S}_m\}_{m=1}^N$, denoted as [74-75]

$$\varphi(x|\theta_m) = (2\pi)^{-d/2}(\|\mathbf{S}_m\|)^{-0.5}.\exp\left\{-\frac{1}{2}(x-\boldsymbol{\mu}_m)^T\boldsymbol{S}_m^{-1}(x-\boldsymbol{\mu}_m)\right\} \quad (3.4)$$

where d=2 is the dimension of the feature space, $\|.\|$ is the matrix determinant. The Gaussian mixture model (GMM) representing the probability density function (pdf) of x is defined as [74-75]

$$p(x|\Theta) = \sum_{m=1}^{N} \alpha_m \varphi_m(x|\theta_m) \quad (3.5)$$

where $N = 3$ is the number of components, $\alpha_m \geq 0$ represents the mixing weight of the component m and $\sum_{m=1}^N \alpha_m = 1$. The unknown parameter set denoted as $\Theta = \{\alpha_m, \boldsymbol{\mu}_m, \boldsymbol{S}_m\}_{m=1}^N$ should be estimated for each Gaussian component using the following EM algorithm's steps. Suppose $\Theta^{\mathrm{r}} = \{\alpha_m^r, \boldsymbol{\mu}_m^r, \boldsymbol{S}_{\boldsymbol{m}}^r\}_{m=1}^N$ be the parameter set in the r-th iteration of the algorithm. The first step of the EM algorithm is to initialize the Gaussian component parameters denoted as $\Theta^1 = \{\alpha_m^1, \boldsymbol{\mu}_m^1, \boldsymbol{S}_{\boldsymbol{m}}^1\}_{m=1}^N$. The centers of the components (i.e., means) are selected by a random initialization technique and the covariance matrix of each component, $\boldsymbol{S}_m^1$, is also initialized as [75]

$$\boldsymbol{S}_m^1 = \left\{\frac{1}{10d}\,\mathrm{trace}\left(\frac{1}{R}\sum_{i=1}^{R}\left(x^i-\boldsymbol{\mu}\right)\left(x^i-\boldsymbol{\mu}\right)^T\right)\right\}\mathbf{I} \quad (3.6)$$

where $\boldsymbol{\mu}$ is defined as the global mean of the feature vectors and $\mathbf{I}$ is the identity matrix. The mixing weights of the components are uniformly initialized by $\{\alpha_m^1 = 1/N\}_{m=1}^N$.

Then, the E- and M-Steps [74] are iteratively employed to re-estimate the parameters $\alpha_m^r, \boldsymbol{\mu}_m^r, \boldsymbol{S}_m^r$ for $r \geq 2$.

E-step: Compute the posterior probability that block x^i belongs to m th component denoted as

$$h_m^r(x^i) = \frac{\alpha_m^{r-1}\varphi(x^i|\boldsymbol{\mu}_m^{r-1}, \boldsymbol{S}_m^{r-1})}{\sum_{m'=1}^{N} \alpha_{m'}^{r-1}\varphi(x^i|\boldsymbol{\mu}_{m'}^{r-1}, \boldsymbol{S}_{m'}^{r-1})} \tag{3.7}$$

M-step: Re-estimate the mean and covariance vectors of each component by using $h_m^r(x^i)$, i.e., parameter optimization:

$$\alpha_m^r = \frac{1}{R}\sum_{i=1}^{R} h_m^r(x^i) \tag{3.8}$$

$$\boldsymbol{\mu}_m^r = \frac{\sum_{i=1}^{R} h_m^r(x^i)x^i}{\sum_{j=1}^{R} h_m^r(x^j)} \tag{3.9}$$

$$\boldsymbol{S}_m^r = \frac{\sum_{i=1}^{R} h_m^r(x^i)(x^i - \boldsymbol{\mu}_m^r)(x^i - \boldsymbol{\mu}_m^r)^T}{\sum_{j=1}^{R} h_m^r(x^j)} \tag{3.10}$$

The EM steps, are iteratively used until the conditional expectation of the log-likelihood function of the GMM denoted by

$$\mathcal{L}(X|\Theta^r) = \sum_{i=1}^{R}\sum_{m=1}^{N} h_m^r(x^i)\log\left[\alpha_m^r\varphi\left(x^i|\boldsymbol{\mu}_m^r, \boldsymbol{S}_m^r\right)\right] \tag{3.11}$$

converges to a local optimum. This convergence is theoretically guaranteed by [74]. A local optimum can be obtained when the following criterion is fulfilled [75]

$$\left|\mathcal{L}(X|\Theta^r) - \mathcal{L}\left(X\middle|\Theta^{r-1}\right)\right| < 10^{-5}\left|\mathcal{L}\left(X\middle|\Theta^{r-1}\right)\right| \tag{3.12}$$

In (3.12), the absolute values are needed as the log-likelihood function can be negative in Eq. (3.11). Fig. 3.4 illustrates clustering results (i.e., partitions) corresponding to different local optima obtained from the EM experiment

initialized by using a random initialization technique [75] and *k*-means clustering algorithm [63] for a low DOF image. To evaluate different partitions, this experiment was individually ran 1000 times at two consecutive levels of block size (e.g., $64 \times 64, 32 \times 32$) for clustering the *butterfly* image. Each partition consists of three cluster labels illustrated by black, gray, and light gray colour for background, uncertain, and sharp blocks, respectively. As shown in Figure 3.4, clustering with a large block size (i.e., low resolution) may result in preserving the most important regions (object blocks) but compromising the details of an object. Conversely, decreasing the block size (i.e., higher resolution) may preserve more details of an object but includes irrelevant high-contrast and low-energy blocks as well. Moreover, the different initialization of cluster centres in the EM algorithm may affect the final partition and consequently produce different results (see Figure 3.4).

3.5 The Proposed Algorithm

The EM algorithm as a local (greedy) method is dependent on initialization process [75]. It is also well known that using different initializations of component variables and focusing on the highest likelihood estimate alone cannot necessarily guarantee to find the best partition (i.e., global optimum). As evidenced by our experiments, the EM algorithm initialized by a re-sampling technique (i.e., random initialization) or standard *k*-means clustering algorithm may converge to different partitions in different runs of the algorithm.

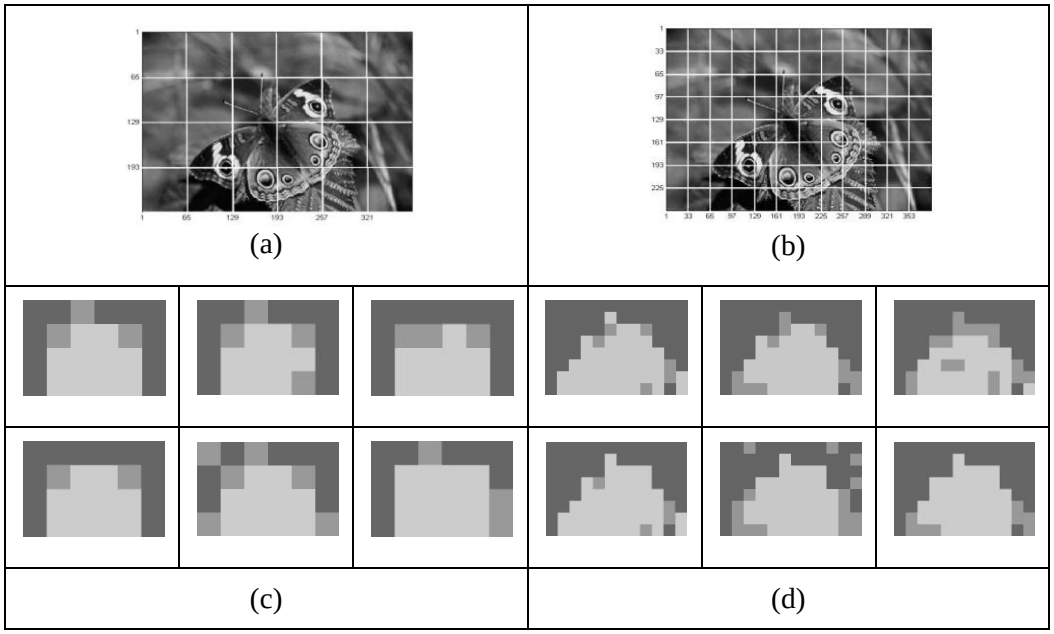


Figure 3.4: Illustration of partitions corresponding to different local optima in the EM algorithm after 1000 iterations. (a) and (b) Grayscale image (namely *butterfly*) with uniform partitioning at level 64×64 and 32×32, respectively. (c) and (d) Different partitions after converging to a local optimum at two consecutive levels. Each partition consists of three region classes: defocused (illustrated by black colour), uncertain (gray colour), and sharp (light gray).

To overcome the above disadvantages and obtain the best partition representing image object(s) in an efficient and robust way at the level of block size, we propose to use the EM algorithm in two consecutive levels of block size (i.e., two resolutions) along with a fusion decision approach. In this sense, we allow the EM algorithm to iteratively reach different local optima based on various initial configurations (i.e., *k*-means and re-sampling) in each level of resolution and create new partitions. For each level independently, these partitions are aggregated based on the final posterior probabilities estimated in the EM algorithm for a certain block. This process is repeated for all blocks in

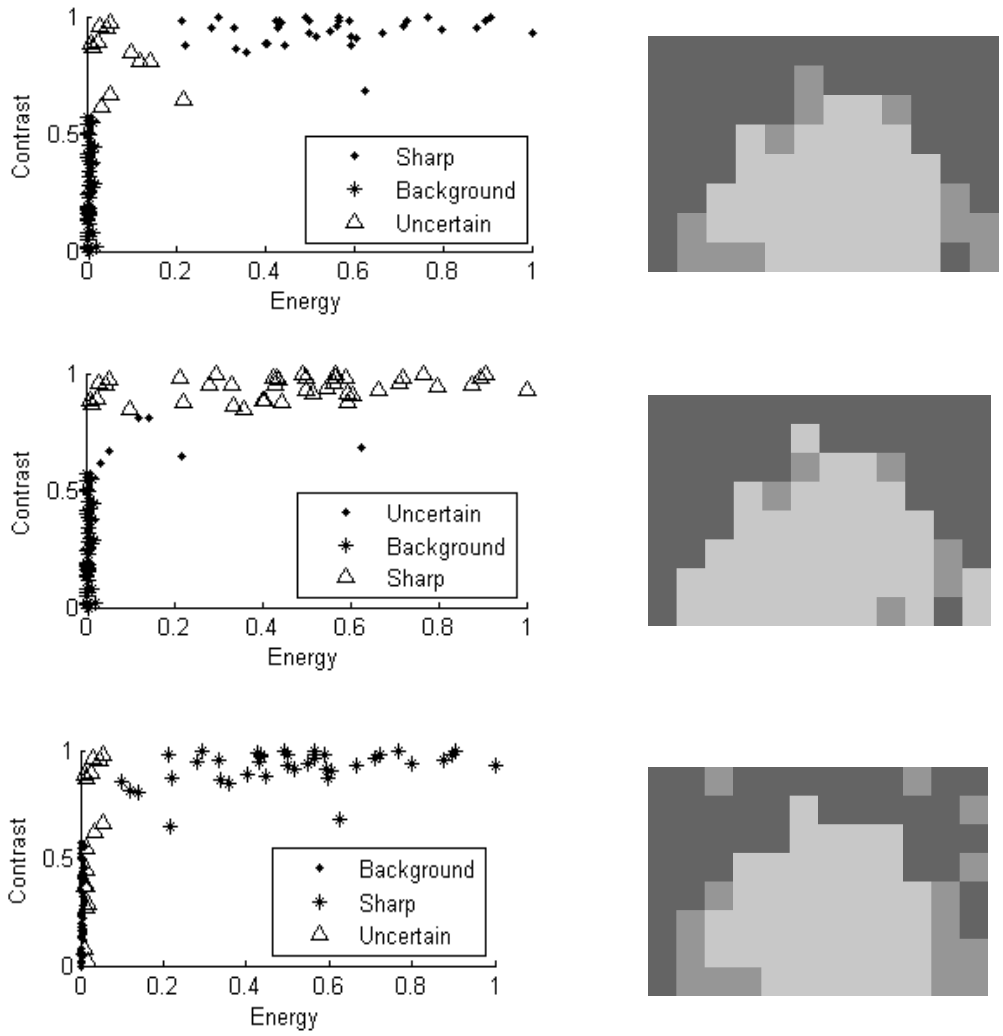
each level. Finally, two obtained partitions are fused by a combining process; this results in a final clustering for a low DOF image at the level of block size. In the following, the details of the fusion decision approach are explained.

3.5.1 Aggregation of Partitions

The basic idea for aggregating a number of partitions is originated from the context of multiple classifier systems (i.e., classifier ensembles) [79-80]. In clustering ensembles [77-78, 81-85], several partitions are aggregated to produce a final clustering of a dataset which is better than each individual clustering result. The goal of clustering ensembles is to improve the robustness and the stability of a clustering process [77, 81]. To achieve this, one needs to address two major difficulties in clustering ensembles including diversity of clustering (i.e., how to generate different partitions) and finding a consensus function (i.e., how to resolve the label correspondence problem and also how to aggregate different partitions). The diversity of clustering can be obtained from several sources to produce different partitions such as using different clustering algorithms [77, 83], different initial configuration [77, 80-81], splitting the original dataset [83], and employing different features [80, 83]. Selecting a consensus function from multiple partitions can be approached from different perspectives such as hyper-graph-partitioning [81], voting approaches [77], quadratic mutual information [78, 83], and co-association matrix [77].

In our approach, an ensemble can be formed as the result of T_1 different runs of the EM algorithm from different initial starting points. To provide diverse partitions, the *k*-means clustering algorithm and a random initialisation technique [75] were employed to initialize the centres of the clusters in two

levels of resolutions. Since the cluster labels are symbolic and any permutation of the symbols corresponds to the same partition, it is necessary to find the correspondence between the cluster labels in a partition. This problem becomes even more difficult when a clustering ensemble approach deals with different numbers of clusters [83]. However, this work utilizes a fixed number of clusters and consequently resolving a correspondence problem is straightforward. In our approach, the cluster that corresponds to the sharp regions (i.e., sharp focused) will have a mean that has the largest Euclidian distance from the centre of the contrast-energy feature space. On the other hand, the cluster that corresponds to the background regions (i.e., out-of-focus) will have the smallest Euclidian distance from the centre of the feature space. This is mainly because sharp and background regions are characterized by distinctive contrast-energy features which is based on the definition of region classes outlined at Section 3.2. Therefore, all three region classes in a partition are assigned by their corresponding labels. Figure 3.5 illustrates the contrast-energy feature space of the *butterfly* image at level two (e.g., 32×32) and three possible partitions of regions.



(a) (b)

Figure 3.5: Illustration of the three possible clustering results (partitions) of 96 blocks for the *butterfly* image at level two (e.g., 32×32). These partitions use different sets of labels (column (a)). Sharp and background components will have the largest and smallest Euclidean distance from the centre of contrast-energy feature space. (b) Corresponding labelled blocks. The sharp, uncertain, and background blocks have been illustrated by the colours light gray, gray and black, respectively.

Assume that a partition is represented as a set of cluster labels which have been assigned by the EM algorithm. Let $\Pi = \{\pi_i\}_{i=1}^{T_1}$ be a set of T_1 partitions obtained from different runs of the EM algorithm in a level. Suppose that the estimates of posterior probabilities for a block in different partitions at a level are given as N-dimensional vectors $[\pi_i^1, \dots, \pi_i^N]^T \in [0,1]^N$, $i = 1, \dots, T_1$, where $\pi_i^j(x^k) = h_j^r(x^k)$ denotes the final estimated posterior probability of the block x^k in j-th component/cluster obtained from (3.7). The T_1 partitions for the block x^k can be organized as the matrix

$$H(x^k) = \begin{bmatrix} \pi_1^1(x^k) & \pi_1^j(x^k) & \pi_1^N(x^k) \\ \vdots & \vdots & \vdots \\ \pi_i^1(x^k) & \pi_i^j(x^k) & \pi_i^N(x^k) \\ \vdots & \vdots & \vdots \\ \pi_{T_1}^1(x^k) & \pi_{T_1}^j(x^k) & \pi_{T_1}^N(x^k) \end{bmatrix} \tag{3.13}$$

The posterior probability values in column j of the matrix H are the individual supports for component j. We use a set of T_1 weights, one for each partition. The weight for a partition is based on the value of log-likelihood estimate in the EM algorithm. The larger the log-likelihood estimate, the more likely the important partition. Suppose the log-likelihood value of $i\text{-}th$ partition is denoted by $\mathcal{L}_i \stackrel{\text{def}}{=} |\mathcal{L}(X|\Theta^r)|$. The weight for $i\text{-}th$ partition can be calculated by $w_i = \frac{\mathcal{L}_i}{\sum_{l=1}^{T_1} \mathcal{L}_l}$, where $\sum_{i=1}^{T_1} w_i = 1$. The overall degree of support for component j can be obtained by a weighted averaging operator [80]

$$\beta_j(x^k) = \sum_{i=1}^{T_1} w_i \pi_i^j(x^k), \quad j = 1, \dots, N \tag{3.14}$$

The final decision is made by using a maximum combination function such that the cluster label of the block x^k is found as the index of the maximum $\beta_j(x^k)$

$$\beta(x^k) = \max_j \{ \beta_j(x^k) \} \tag{3.15}$$

Therefore, a new partition at a level can be obtained by repeating the above procedure for all the blocks in an image. In Figures 3.6(a) and (b), the final partitions using aggregating different clustering results at the first and second levels have been illustrated. As expected, the object blocks of the image are correctly labelled as sharp blocks illustrated by light gray colour at the first level (see Figure 3.6(a)). Moreover, the final partition at the second level mostly includes more details of the object along with a number of uncertain blocks which are originally related to the defocused areas. Hence, an approach is necessary that can consolidate cluster labels from the two consecutive levels and provide a partition which represents an image object.

3.5.2 Combining Partitions at Two Consecutive Levels

In our approach, by using two partitions obtained from our ensemble clustering approach at two consecutive levels of block size and a combining process, we aim to achieve a reliable partition at the level of block size. For convenience, we call a block at the low resolution (e.g., 64×64) as a parent block and its four subdivision blocks at the higher resolution (e.g., 32×32), which are at the same spatial location, as child blocks (Figure 3.6(c)). As illustrated in Figure 3.7, a parent or child block in a partition can accept one of the three existing cluster labels: sharp, uncertain, and background. By evaluation of 100 low DOF images at the level of block size, we found that if the cluster label of a parent block is sharp or uncertain, the clustering process at

the level two is able to identify the true cluster labels of its child blocks. In other words, the cluster label of a child block remains unchanged when its parent cluster label is sharp or uncertain in combining two partitions. For a parent block with a background cluster label, if the cluster label of a child block is sharp and at least one of its 4-connectiveity neighbours (i.e., horizontal and vertical) has been labelled as sharp, the label of this block is changed to uncertain in combining partitions; Otherwise, the cluster label is change to background. Figure 3.8 illustrates the aggregating and combining processes for the *butterfly* image.

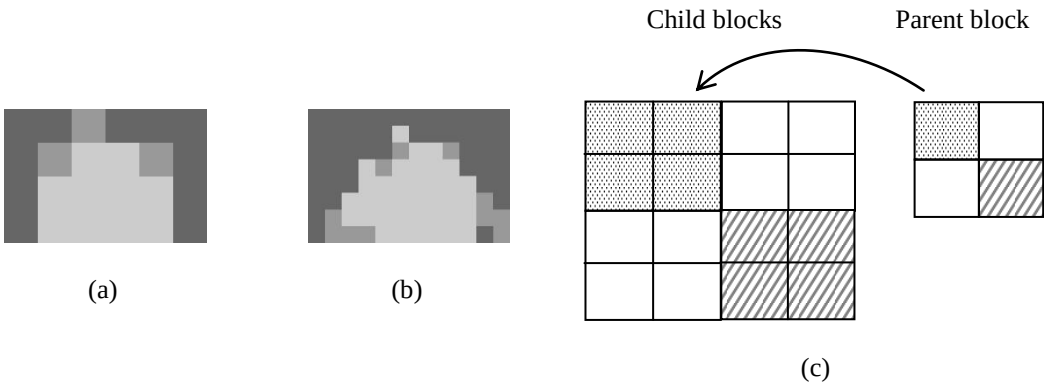


Figure 3.6: (a) and (b) Final clustering results obtained from the aggregation of partitions at two consecutive levels, respectively. (c) Illustration of a parent block and its subdivision blocks as child blocks.

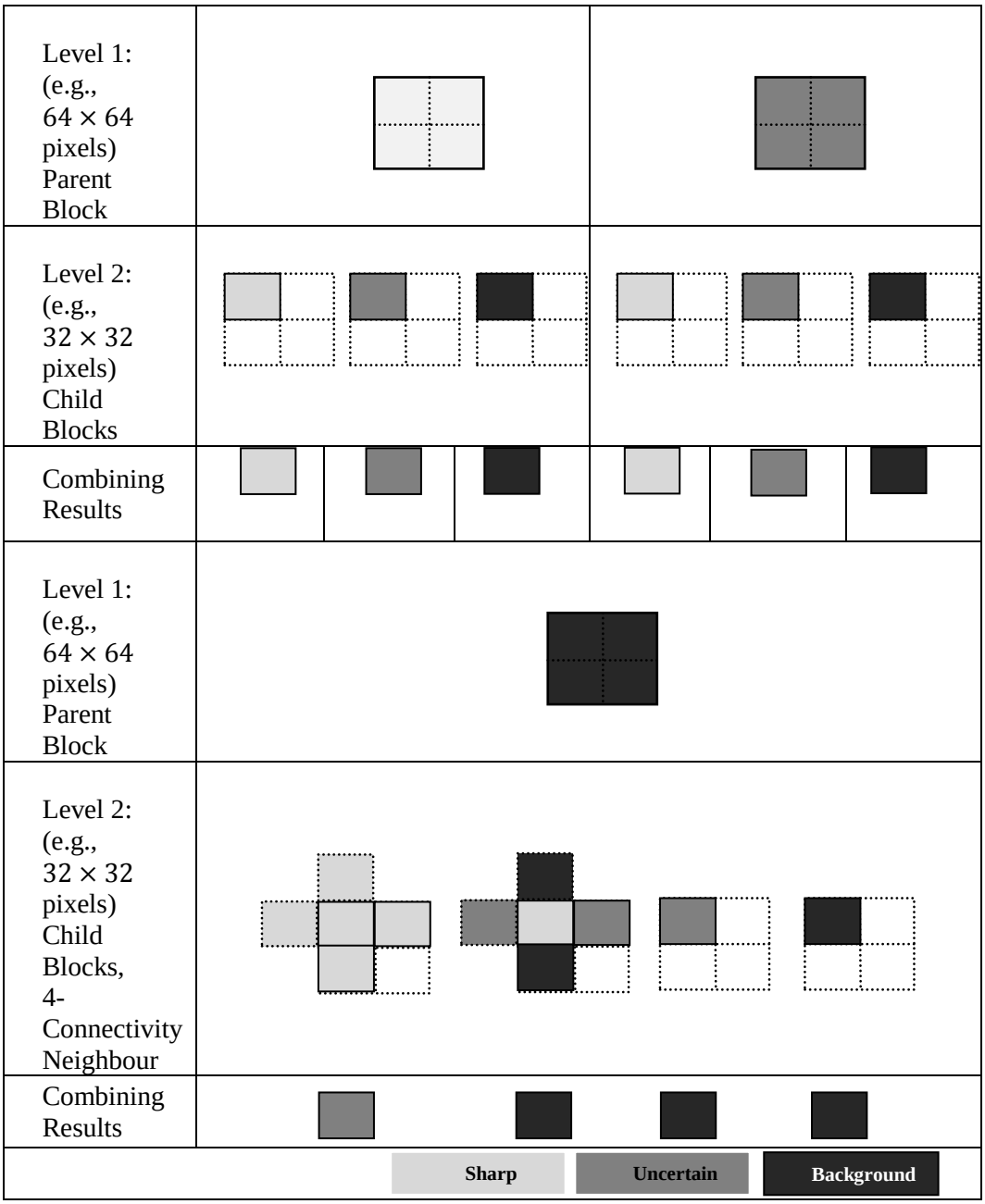


Figure 3.7: Illustration of combining the blocks of two consecutive levels.

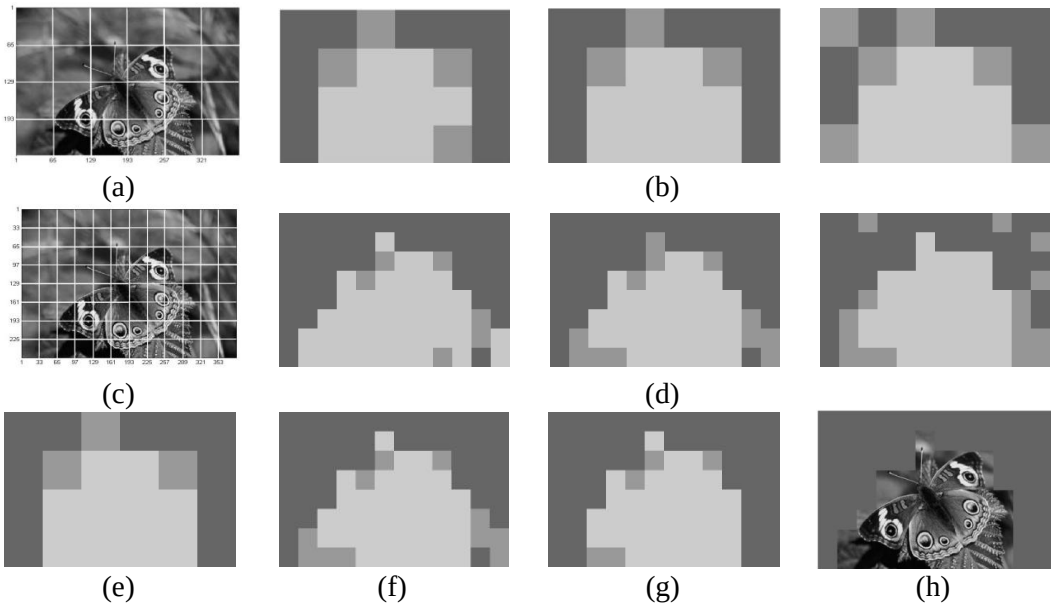


Figure 3.8: Illustration of various partitions and the fusion decision process. (a) and (c) Grayscale images with uniform partitioning at two consecutive levels, i.e., 64×64 and 32×32. (b) and (d) Various partitions corresponding to different local optima at the first and second level, respectively. (e) and (f) partitions after aggregating process in each level. (g) Final partition after combining (e) and (f). (h) Clustered image.

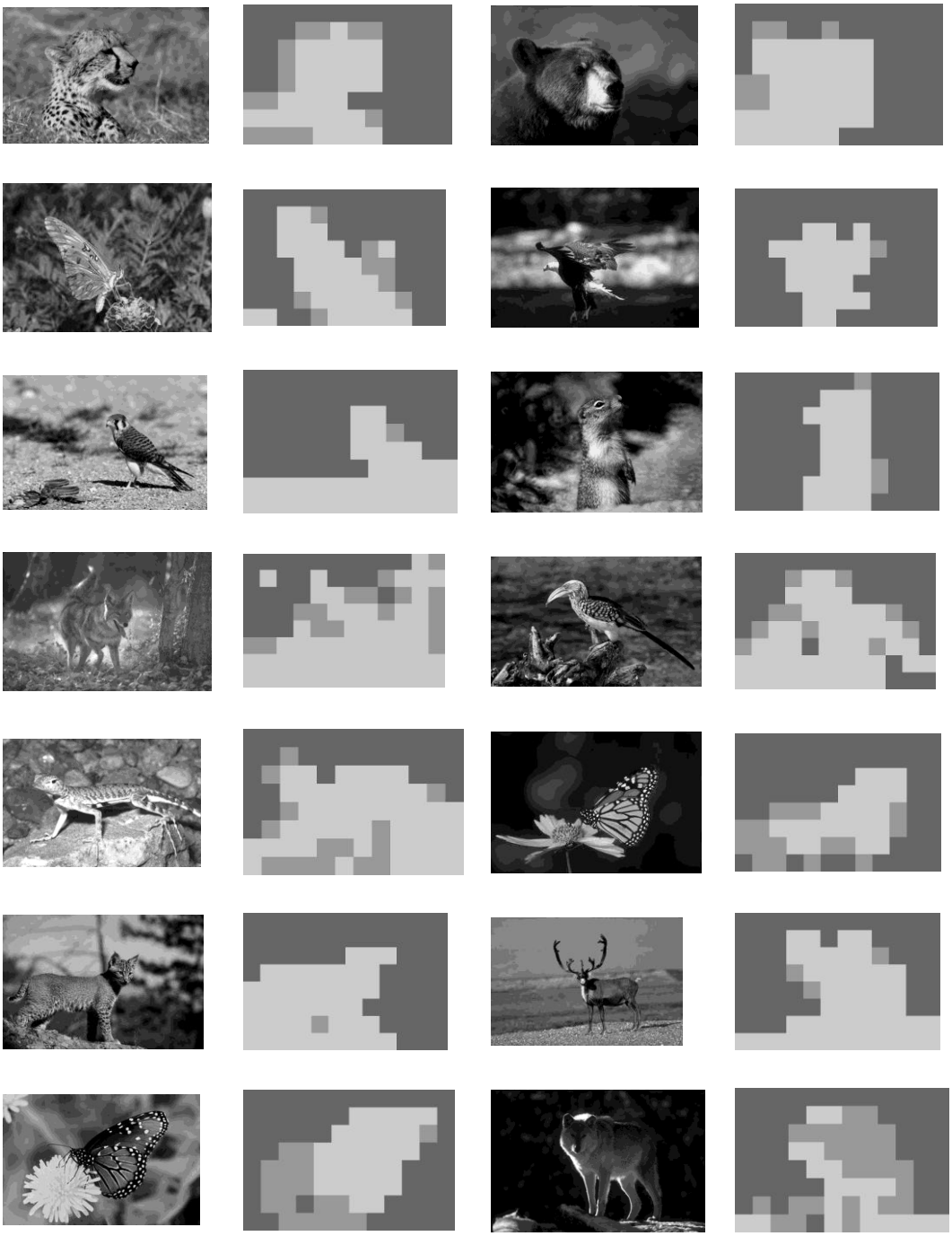
An algorithmic outline of the ensemble EM clustering is provided below. The input is a low DOF image in any format (e.g., JPEG, GIF, etc.). The grayscale format of the original image is used in this algorithm. The output of the algorithm is interest regions at the level of block size. We empirically found that $T_1 = 10$, the number of ensemble members, was more than a sufficient number of iterations to reach different local optima in all experiments.

Algorithm Ensemble EM Clustering

1. Select the number of ensemble members T_1 (number of iteration) and the number of cluster N
2. Initialize the level and size of resolution: $L \leftarrow 1$, $S_L \leftarrow s \times s$, where $s \leftarrow 2^{\lfloor \log_2(0.25 \times \max(n_{Row}, n_{Col}) \rfloor}$, and n_{Row}, n_{Col} denote the number of pixels in row and column of a given image, respectively.
3. Divide the image into R_L uniform blocks with the size of S_L and extract the local contrast and energy of the blocks as data points: $x^k = (x_c^k, x_e^k)$, $k = 1..R_L$
4. $i \leftarrow 1$
5. Cluster data points at the level of L to find partition π_i as follow:
    a) while $(i < T_1)$
        i. Apply the EM algorithm's steps to find a local optimum, i.e., a new partition
        ii. Add partition π_i to the ensemble $\Pi = \{\Pi, \pi_i\}$
        iii. $i \leftarrow i + 1$
6. Aggregating process on Π to find partition P_L^*
7. $L \leftarrow L + 1$, $s \leftarrow \frac{s}{2}$, $S_L \leftarrow s \times s$
8. If $L < 2$, go to step 3
9. Combining partitions $\{P_L^*\}_{L=1}^2$
10. Return the final partition

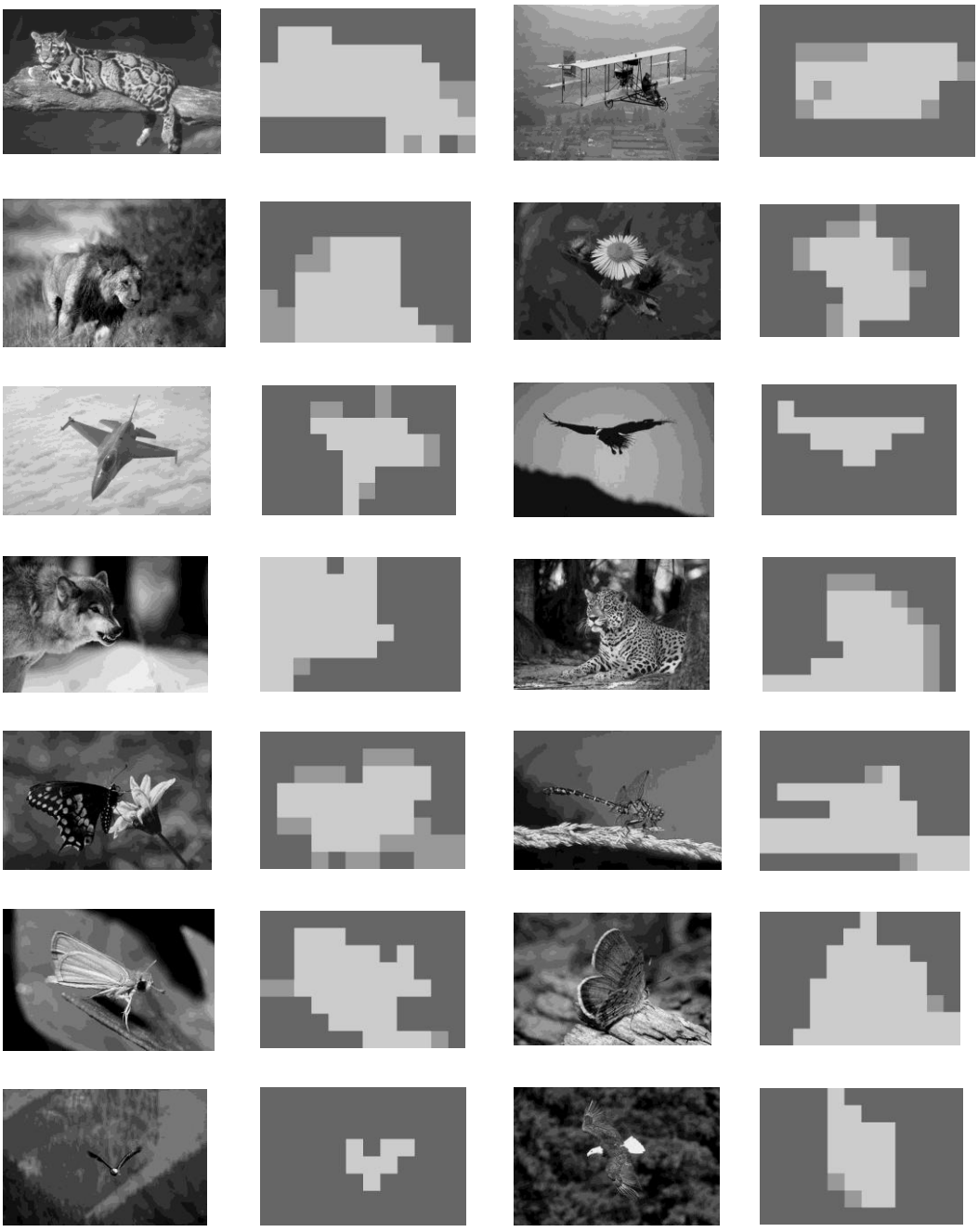
3.5.3 Clustering Results

In this section, a set of partitions as the results of the proposed algorithm is presented. A set of images from Corel dataset [56-57, 70] have been selected to test the algorithm. As shown in Figure 3.9-3.11, each clustering result (or partition) includes three region classes: defocused, uncertain, and sharp areas illustrated by black colour, gray colour, and light gray, respectively. As it can be seen from the results, the algorithm is able to effectively separate background areas from the objects in foreground at the level of block size.



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

Figure 3.9: Illustration of final partitions for a number of images obtained from the algorithm with $T_1 = 10$. (a) and (c) Grayscale test images. (b) and (d) Final partitions after employing the combining process.



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

Figure 3.10: Illustration of final partitions for a number of images obtained from the algorithm with $T_1 = 10$. (a) and (c) Grayscale test images. (b) and (d) Final partitions after employing the combining process.

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

Figure 3.11: Illustration of final partitions for a number of images obtained from the algorithm with $T_1 = 10$. (a) and (c) Grayscale test images. (b) and (d) Final partitions after employing the combining process.

3.6 Summary

In this chapter, a novel ensemble clustering algorithm for low DOF images has been presented. Existing clustering techniques are incapable of reliable clustering due to the dependency on an initialisation process. The proposed ensemble clustering algorithm is developed to extract meaningful information at the level of block size. One of the advantages of the proposed algorithm is that it considers the three types of regions including out-of-focus (i.e., background), sharp focused, and uncertain. Therefore, if a block of ROI includes partially sharp focused and smooth regions, the algorithm is successful in extraction the object. Moreover, block-wise processing in this stage makes the algorithm very time efficient compared with the methods which employ whole image pixels of a low DOF image.

Chapter 4

4. PIXEL-BASED ROI EXTRACTION APPROACHES

4.1 Introduction

In this chapter, two approaches for extracting ROI are introduced. The first approach, which is the main focus of this Chapter, aims to create a binary saliency map of all smooth and focused regions from the block-based interest regions in a clustered image. To achieve this, we have developed a new methodology by optimising a threshold in a DOG image and by using morphological operations. The second approach benefits from the advantages of a graph cuts segmentation algorithm. In this approach, the ensemble clustering algorithm imposes certain hard constraints for segmentation by indicating certain pixels that have to be part of the object and certain pixels that have to be part of the background. The rest of the image is segmented by computing a

global optimum among all segmentations satisfying the hard constraints. Segmentation results for both approaches are also provided using images selected from Corel dataset and the Web.

4.2 Extracting Interest Regions at the Level of Pixel by Determining Optimum Threshold

In the previous chapter, we segmented an image into three classes of regions at the level of block size (e.g., 32×32 for image size of 384×256 or 256×384). As shown in Figure 3.9-3.11, the uncertain blocks illustrated by gray colour in the final clustering results are mostly related to the object regions (i.e., interest regions) which have low intensity changes, partially sharp properties, and smooth boundaries. Therefore, the blocks with sharp and uncertain cluster labels are integrated and considered as block-based interest regions (i.e., ROI blocks).

In order to extract pixel-based interest regions from the block-based regions, an approach needs to be employed to augment the visibility of low intensity variations of the regions and boundaries while reducing the effects of noise which originates from the block-based grouping. To address this problem, we have developed a new methodology by optimizing a threshold in a DOG image and by using morphological operations. The main focus of this methodology is to create a binary saliency map of all smooth and focused regions from the block-based interest regions in a clustered image.

4.2.1 DOG and Binarization Functions

To detect intensity variations present in the block-based interest regions, the DOG is used [86-92]. This function, which is the approximation of

Laplacian of Gaussian, is computationally efficient and can be suitably constructed by subtracting two Gaussians with different standard deviations (i.e., filter scales) given by [91]

$$g(x,y) = G(x,y,\sigma_1) - G(x,y,\sigma_2) \qquad \text{where} \quad \sigma_1 > \sigma_2 \tag{4.1}$$

where $G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$. The filter scales σ_1 and σ_2, which control the thickness property of edges, were set to 0.8 and 0.5, respectively. It has been shown that this function is able to satisfactory detect intensity variations in a region when the filter scales are selected from the ratio 1:1.6 [90]. We filtered a clustered image including ROI blocks using the DOG function and obtained a DOG image denoted by

$$D(x,y) = g(x,y) * I_{Clustered} \tag{4.2}$$

where $I_{Clustered}$ denotes the ROI blocks and $*$ is the convolution operation in x and y.

The DOG image pixels are thresholded with an intensity value z, i.e., $I^z = \{D(x,y): D(x,y) > z\}$; we will discuss the choice of z shortly. Then, the image I^z is converted to a binary image based on a global thresholding technique which is defined by Otsu's method [53]. The Otsu's method assumes that the gray-level histogram of a given image is bimodal, i.e., the image includes object and background classes. This method was experimentally found to be the most efficient global thresholding method amongst others [54]. However, this assumption is not critical for our method because we aim to find that value of z which best suits the visibility of the interest regions in an image. Figure 4.1 shows the results obtained from the DOG and binarization functions

for the ROI blocks while using a same threshold z in different images. As the images show, the objects that include regions and boundaries which are highly smooth are not correctly recognized. The reason for this is that the same threshold z has been used in DOG images for the different types of images. An elegant solution to this is to define a content-based threshold, which is able to extract the low intensity variation details of the object based on the degree of smoothness in a low DOF image.

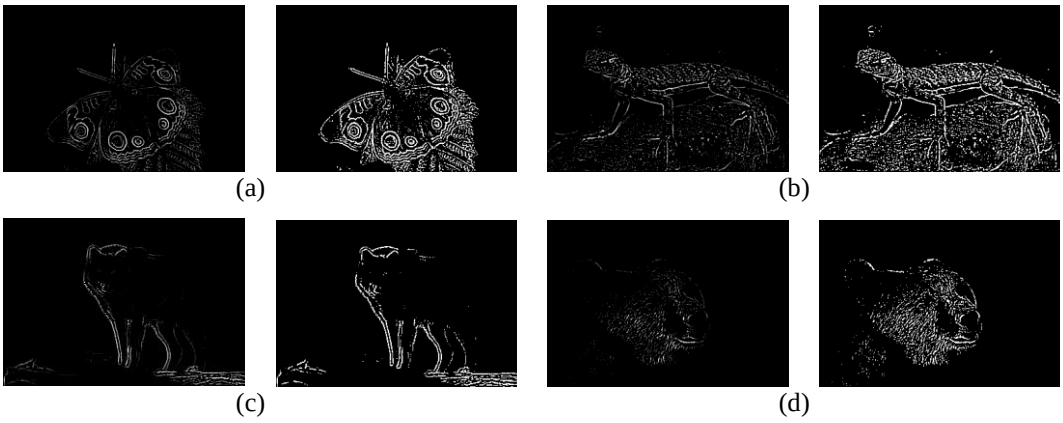


Figure 4.1: Illustration of DOG (left) and corresponding binary images (right) with a same threshold z. (a) An image includes mostly closed boundaries. (b) Image includes noisy regions in the block-based interest regions. (c) and (d) Highly smooth region and boundary

4.2.2 Determining Optimal Threshold

The threshold z determines the sensitivity of the DOG function for extracting the details of intensity variations in ROI blocks. In general, let us denote all ROI and background binary blocks by Ω_R and Ω_B, respectively, where $\Omega_R = \{q_\eta : q_\eta \in \Omega, \eta = 1..N_r\}$, $\Omega_B = \{q_\zeta : q_\zeta \in \Omega, \zeta = 1..N_b\}$, and $\Omega = \Omega_B \cup \Omega_R$ is referred to as all binary blocks in an image obtained from the

DOG and binarization functions. For convenience, we denote a pixel with the value 'one' or 'zero' by white or black pixel in a binary image, respectively. Suppose every binary image is characterized by two functions $\Gamma, \Psi: \mathrm{z} \rightarrow [0,1]$, $z \in [z_1, z_2]$, where z_1 and z_2 denote the minimum and maximum intensity values of the image $D(x, y)$ obtained from (4.2), respectively. We define the function $\Gamma(z)$ as the ratio of retrieved high-energy ROI blocks to the total number of ROI blocks including sharp and uncertain cluster labels in a clustered image as shown in (4.3). The value of this function represents the strength of visibility of the ROI areas so that the higher value shows the more visible regions. We considered an ROI block as high-energy if the total number of white pixels in that block is more than 5% of the total area of the block as shown in (4.4). For instance, a high-energy block with 32×32 pixels covers at least 50 white pixels.

The function $\Gamma(z)$ is denoted by

$$\Gamma(z) = \frac{1}{N_r} \sum_{\eta=1}^{N_r} F_1(z, q_\eta), q_\eta \in \Omega_R \tag{4.3}$$

$$F_1(z, q_\eta) = \begin{cases} 1, & if \quad \sum_{i=1}^{s} \sum_{j=1}^{s} I_{binary}^{z}\left(x_{q_\eta}^{(i)}, y_{q_\eta}^{(j)}\right) > 0.05 \times s^2 \\ 0, & otherwise \end{cases} \tag{4.4}$$

where $I_{binary}^{z}\left(x_{q_\eta}^{(i)}, y_{q_\eta}^{(j)}\right)$ represents the value of pixel (i, j) in an ROI block q_η of the binary image, and N_r is the number of ROI blocks.

The busy-texture properties of the background blocks (background noisy blocks) are also considered for extracting the details of intensity variations in the ROI blocks. The function $\Psi(z)$, representing the noisy characteristics of the background blocks, is defined as the number of retrieved busy-texture blocks in background areas to the total number of background blocks as shown in (4.5).

We also experimentally found that a background noisy block is mostly visible when there are more than 0.5% white pixels in the block total area according to (4.6). For example, a block with 32×32 pixels is assumed as a noisy block when it has more than 5 white pixels.

$$\Psi(z) = \frac{1}{N_b}\sum_{\zeta=1}^{N_b} F_2(z, \hat{q}_\zeta),\ \hat{q}_\zeta \in \Omega_B \tag{4.5}$$

$$F_2(z, \hat{q}_\zeta) = \begin{cases} 1 & if \quad \sum_{i=1}^{s}\sum_{j=1}^{s} \hat{I}^{z}_{binary}\left(x^{(i)}_{\hat{q}_\zeta}, y^{(j)}_{\hat{q}_\zeta}\right) > 0.005 \times s^2 \\ 0 & otherwise \end{cases} \tag{4.6}$$

where $s^2 = s \times s$ denotes the smallest block size in the algorithm (e.g., 32×32). The value of pixel (i, j) in a background block $\hat{q}_\zeta$ of the binary image is represented by $\hat{I}^{z}_{binary}(x^{(i)}_{\hat{q}_\zeta}, y^{(j)}_{\hat{q}_\zeta})$.

To determine the content-based threshold, the following optimization problem is defined and solved:

$$z^* = \arg \max_{\substack{z\in[z_1,z_2] \\ \Gamma(z)>\Psi(z)}} \{\Gamma(z) - \Psi(z)\} \tag{4.7}$$

The optimization problem in (4.7) suggests that the optimal value of z , z^*, should maximize the percentage of high-energy blocks in the binary image which are matched with the ROI blocks in the clustered image and simultaneously minimize the effect of noise in these blocks. Accordingly, a reduction in the intensity of noisy regions is shown in Figure 4.2(a) and (b) and an increase in the intensity of smooth regions of the ROI blocks have been achieved in Figure 4.2(c) and (d). This results in effective identification of boundary and shape of objects in morphological operations. The effect of

changing threshold z on the performance of the proposed approach is also discussed in Section 5.2.3.

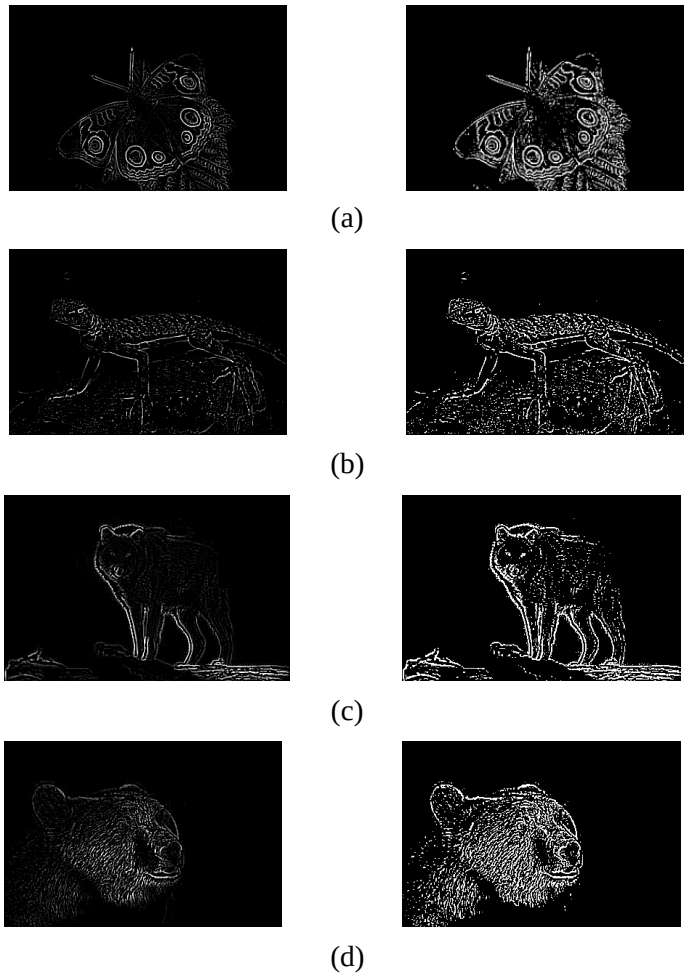


Figure 4.2: Illustration of DOG (left) and corresponding binary images (right) with optimal threshold.

4.2.3 Morphological Processing

In order to identify the underlying region shape and the boundary of an object, morphological operations are employed. These local pixel transformations are able to efficiently extract the form and structure of an object and also eliminate noisy regions in a binary image. We make use of a series of

masks as structuring elements along with dilation, erosion, and filling operations [49, 94] to construct the ROI in a low DOF image.

In the previous Section, we constructed the optimized gray-level DOG image covering focused and smooth regions from the clustered image. To create an RSM from a gray-level DOG image, we firstly employ the dilation and erosion operations (i.e., close operation with a disk structuring element) into the DOG image. Then, the result is converted into a binary image by using the Otsu's method [53] (see Figure 4.3). The obtained RSM is viewed as the initial set for the following dilation operation [49]:

$$P = (RSM \boxplus M_1) \tag{4.8}$$

where $\boxplus$ is a dilation operator and $[M_1(i,j) = 1]_{3\times3}$ is a structuring element. The majority operation [93] is then applied to the dilated image for filling small holes in boundaries as follows:

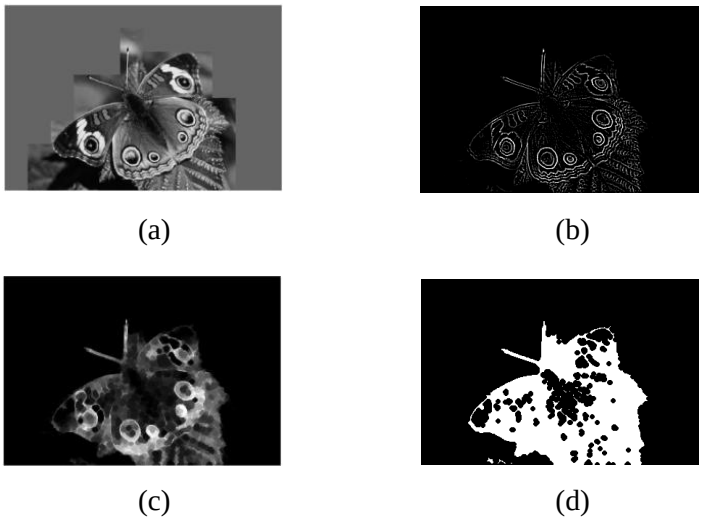
$$P(x,y) = \begin{cases} 1 & \sum_{n\in N_8} P(x_n, y_n) \geq 5 \\ 0 & else \end{cases} \tag{4.9}$$

where N_8 is the set of eight-connectivity neighbor pixels with respect to the origin pixel (x, y). The close operation [49, 93] with a disk shape structuring element is the next step in this methodology to smooth the contours of the object and fuse short gaps between regions defined as

$$(P \boxplus M_2) \boxminus M_2 \tag{4.10}$$

where $\boxminus$ is an erosion operator and M_2 is a circle mask with 4 pixels in its radius. The close operation provides a closed region for the filling process. To extract the boundary, firstly, the interior pixels and then the small and

disconnected objects which have areas less than $\omega = 32$ pixels (for an image size of 384×256 or 256×384) are removed. Finally, the estimated mask (i.e., M_{Est}) is created using the closing and filling operations. Figure 4.4 shows experimental results from each morphological operation.



(a) (b)

(c) (d)

Figure 4.3: Illustration of RSM construction from a clustered image. (a) Clustered image. (b) DOG image obtained by using optimal threshold. (c) Closed image after employing dilation and erosion operations on (b). (d) RSM result after employing Otsu's method [53] on (c).

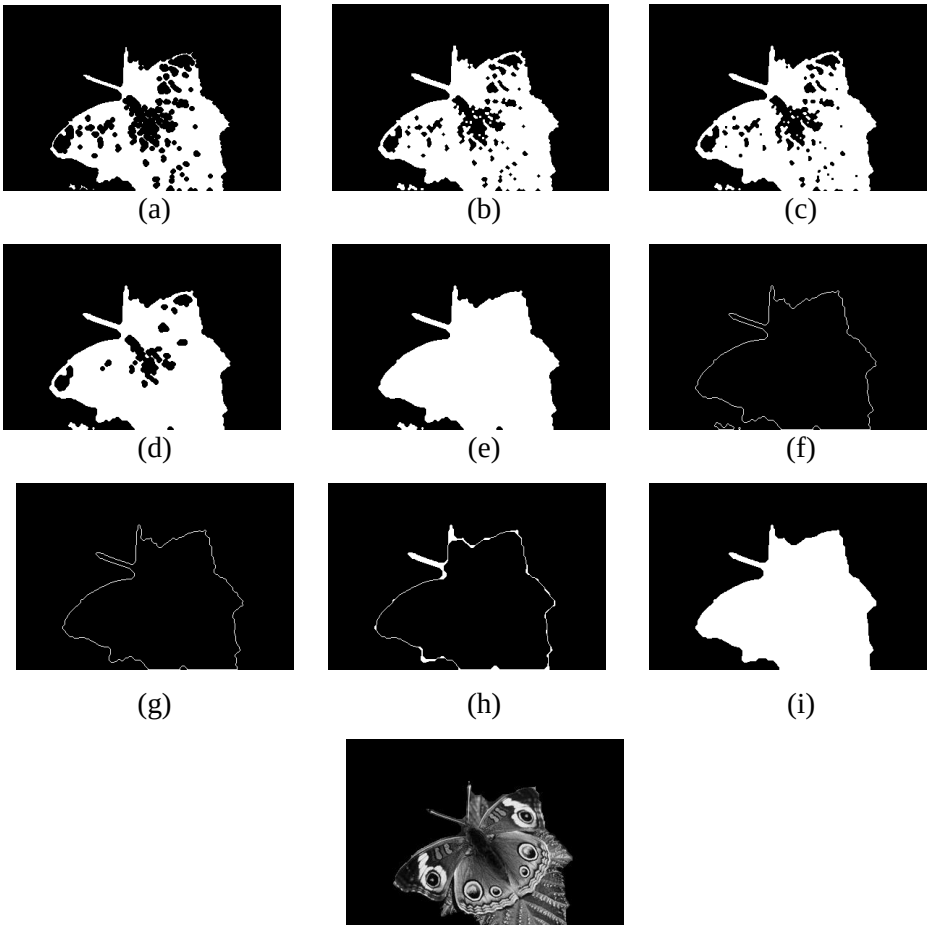
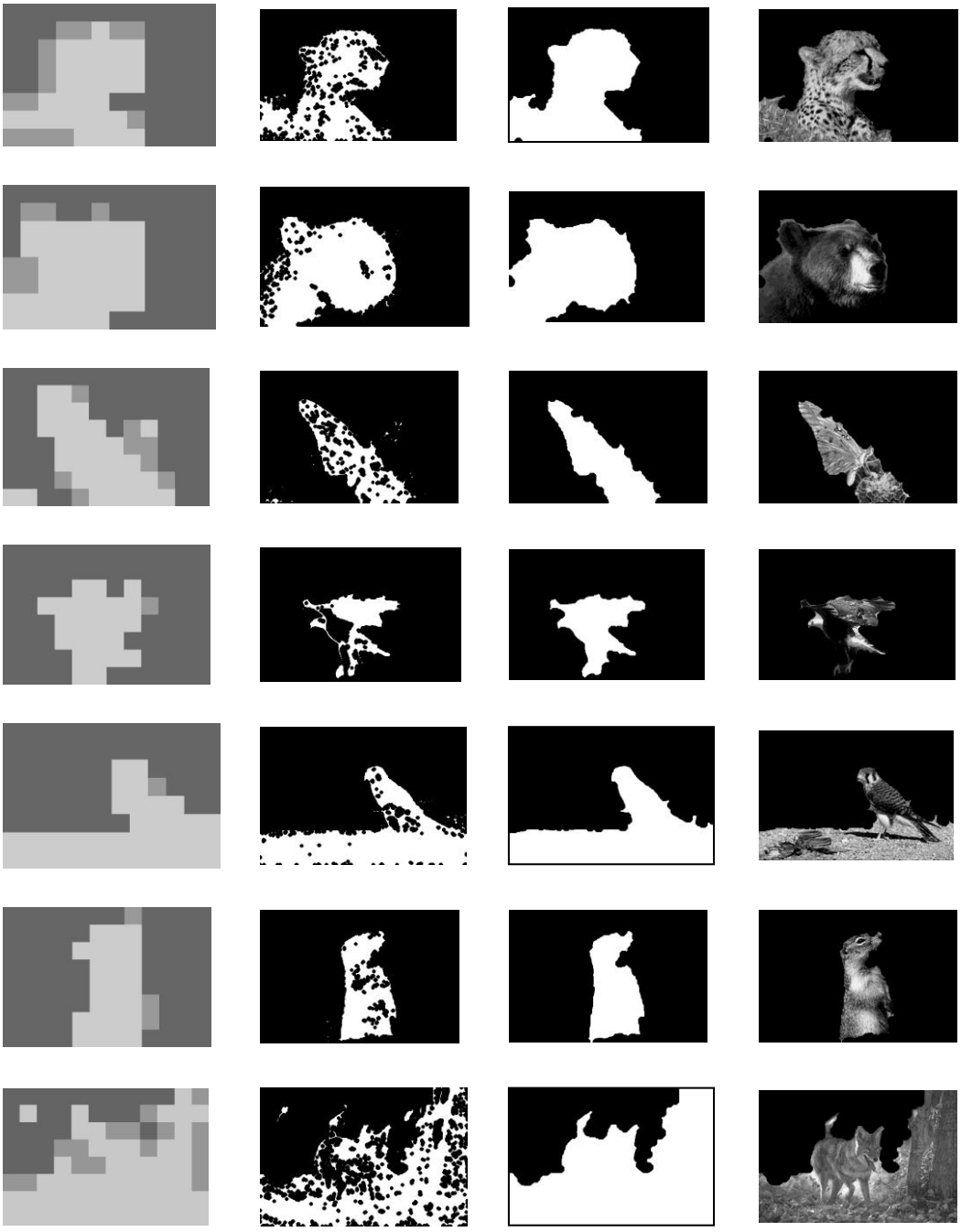


Figure 4.4: Experimental results from each morphological operation. (a) RSM. (b) Dilated image. (c) Majority operation. (d) and (e) Closing and filling processes. (f) Removing interior pixels for boundary extraction. (g) Removing small and disconnected objects covering an area less than $\omega = 32$ pixels. (h) Filling gaps with closing operation. (i) Filling operation to obtain the estimated mask. (j) Interest regions.

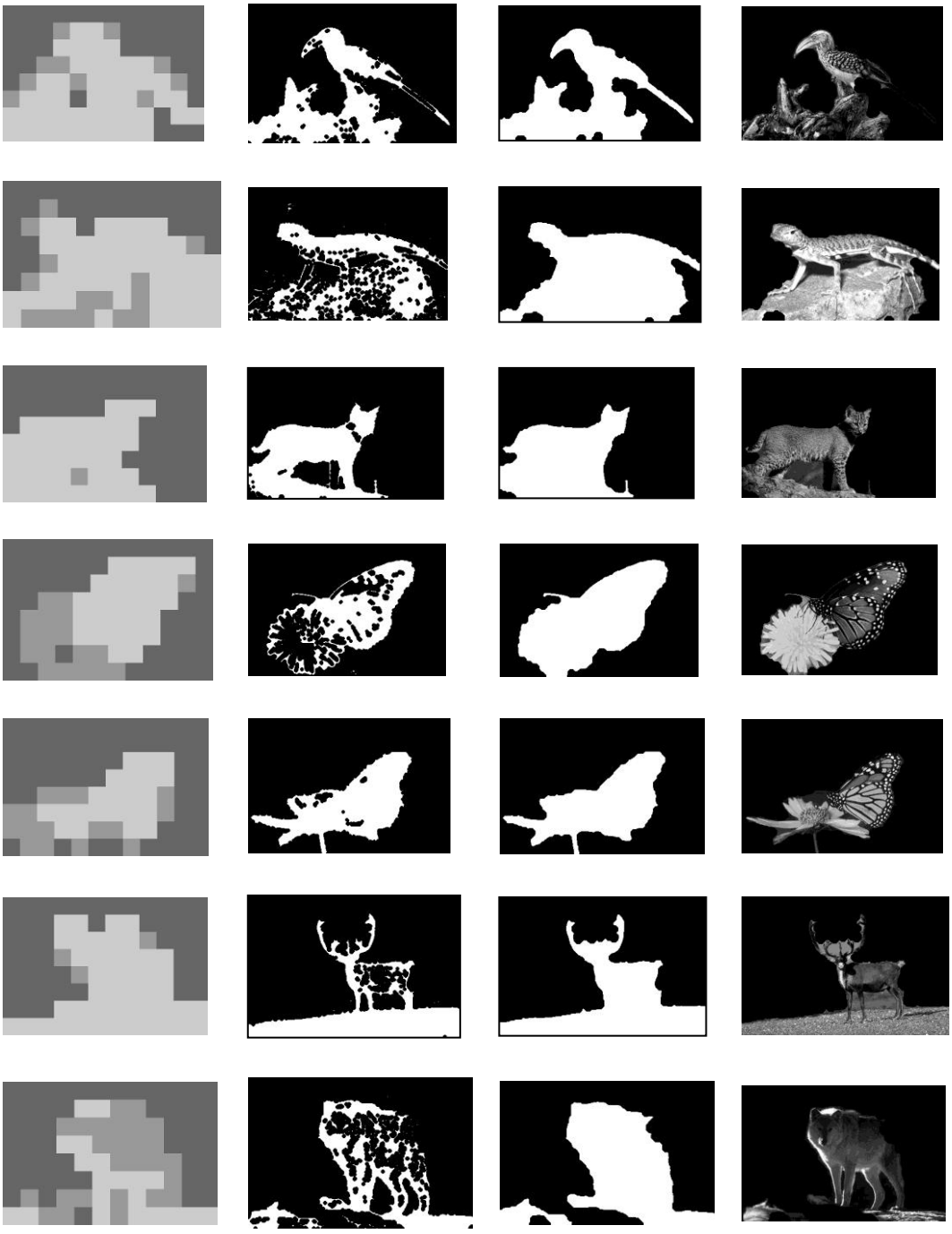
4.2.4 Experimental Results

This Section provides experimental results obtained from the DOG optimisation and morphological operations. We tested the proposed optimisation technique over a number of selected images from the Corel dataset. Figures 4.5-4.10 illustrate these results.



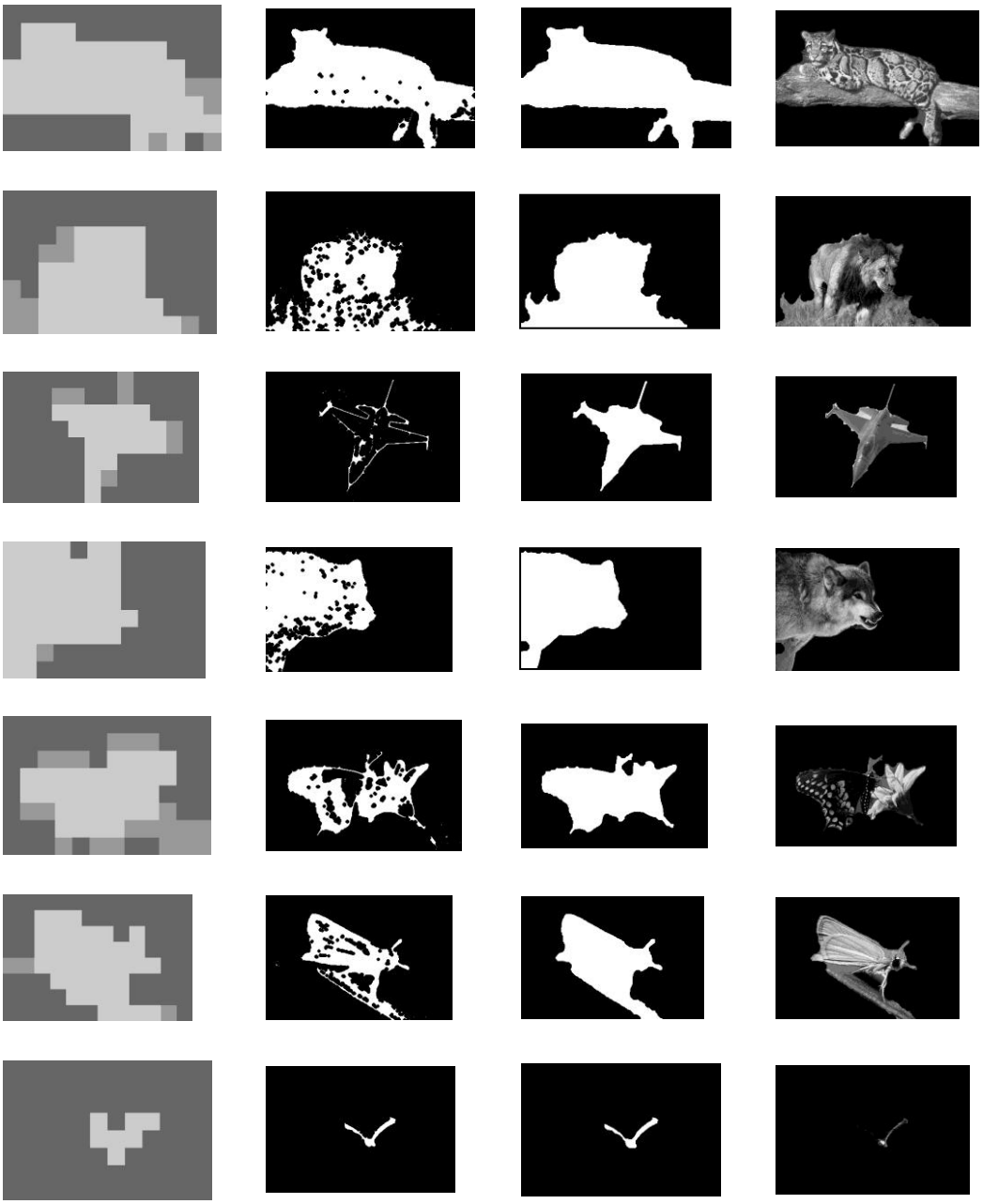
(a) (b) (c) (d)

Figure 4.5: Illustration of final clustering results. (a) Block-based interest regions using the ensemble clustering algorithm. Region saliency map (b). Estimated mask by the proposed approach (c). Final segmentation results (d).



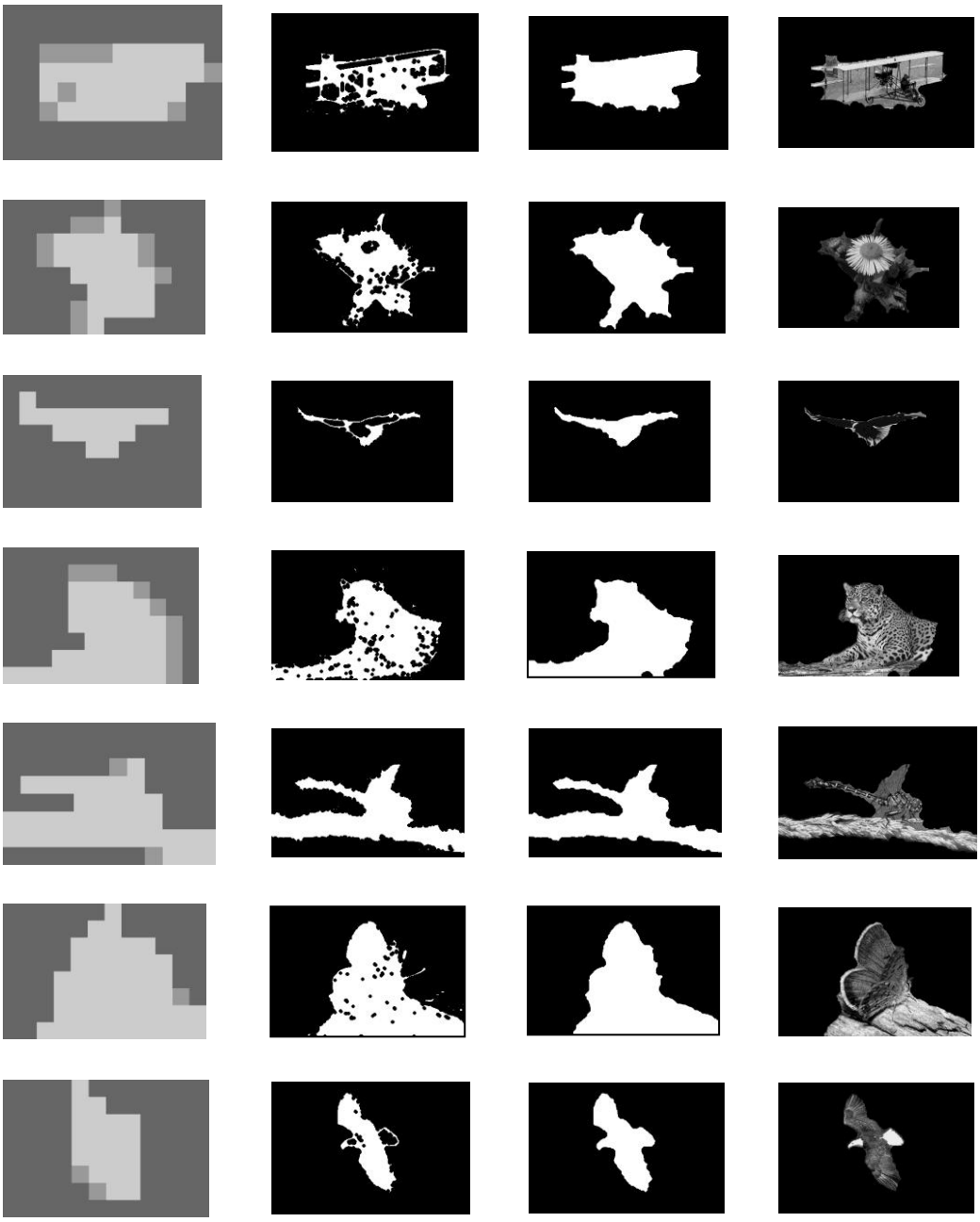
(a) (b) (c) (d)

Figure 4.6: Illustration of final clustering results. (a) Block-based interest regions using the ensemble clustering algorithm. Region saliency map (b). Estimated mask by the proposed approach (c). Final segmentation results (d).



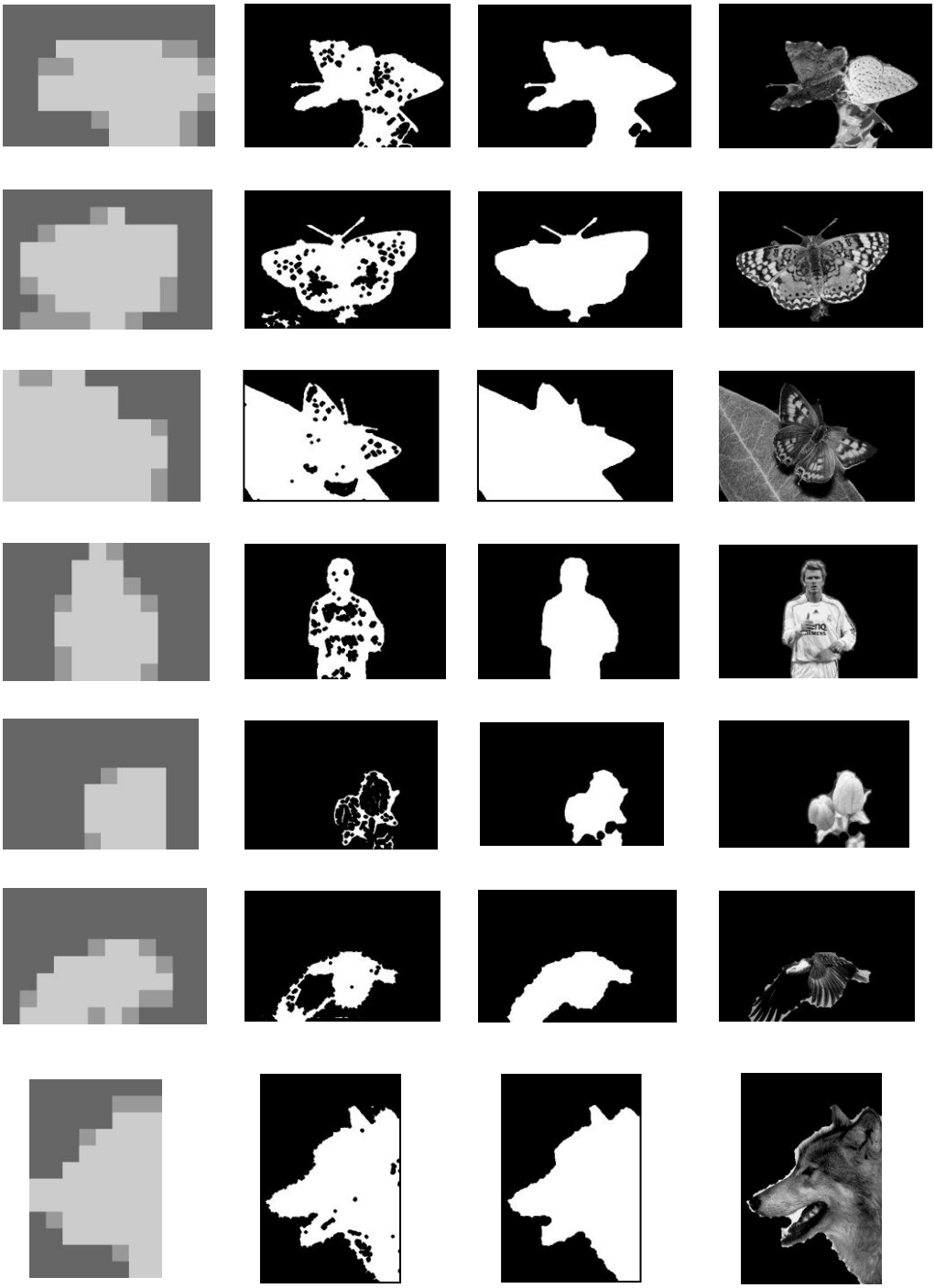
(a) (b) (c) (d)

Figure 4.7: Illustration of final clustering results. (a) Block-based interest regions using the ensemble clustering algorithm. Region saliency map (b). Estimated mask by the proposed approach (c). Final segmentation results (d).



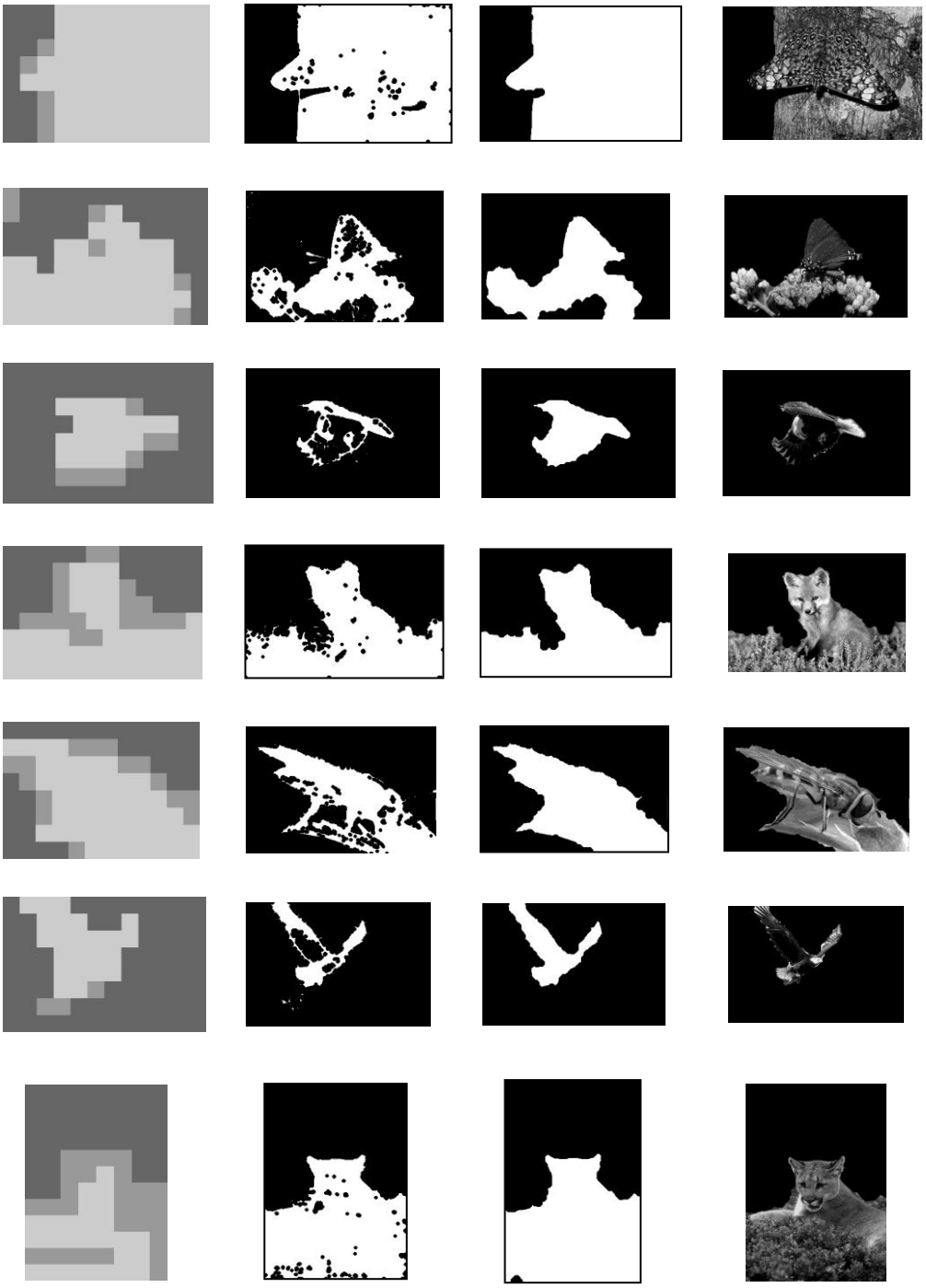
(a) (b) (c) (d)

Figure 4.8: Illustration of final clustering results. (a) Block-based interest regions using the ensemble clustering algorithm. Region saliency map (b). Estimated mask by the proposed approach (c). Final segmentation results (d).



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

Figure 4.9: Illustration of final clustering results. (a) Block-based interest regions using the ensemble clustering algorithm. Region saliency map (b). Estimated mask by the proposed approach (c). Final segmentation results (d).



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

Figure 4.10: Illustration of final clustering results. (a) Block-based interest regions using the ensemble clustering algorithm. Region saliency map (b). Estimated mask by the proposed approach (c). Final segmentation results (d).

4.3 Extracting Interest Regions at the Level of Pixel by Colour-Based Graph Cut Modelling

In this Section, we utilise the graph-cut technique [33] along with colour information to extract interest regions at pixel level. This technique which belongs to energy-based optimisation approaches has already shown a great potential for solving many problems in computer vision and graphics [33, 95-96]. Despite its simplicity, it benefits from the best features of combinatorial graph cuts methods in computer vision such as global optima, practical efficiency, and numerical robustness [97]. Figure 4.11 illustrates the schematic of the proposed approach for segmenting a colour low DOF image into the ROI and background regions by using ensemble clustering and graph cut optimisation approaches. As it can be seen from the Figure, block-based interest regions are identified using the proposed ensemble clustering algorithm presented in Chapter 3. The certain pixels (seeds) of the object and background blocks are used as a topological constraint for the graph cut module. This constraint which is a prior knowledge about image pixels can be used to reduce the search space of feasible segmentation and makes an algorithm time-efficient. In this module, a minimal graph cut is constructed using object and background seeds, which is based on the max-flow method [95, 98], and as a result a corresponding pixel-based interest region is achieved.

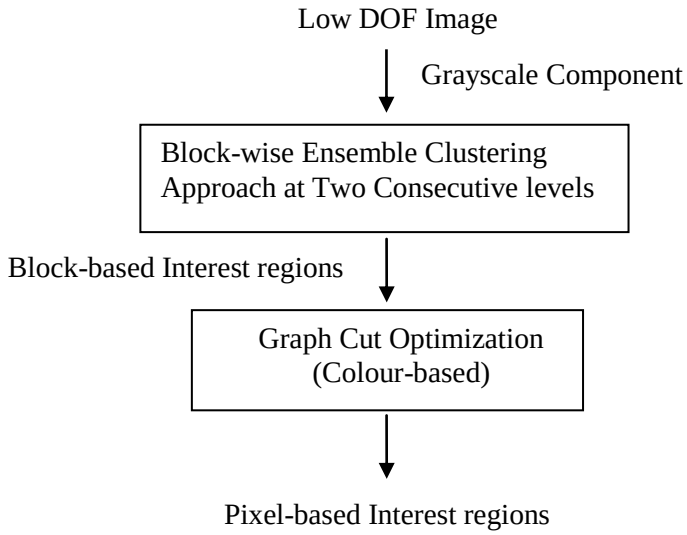


Figure 4.11: The schematic of the proposed approach.

4.3.1 Graph Model Construction and Binary Segmentation

In order to extract pixel-based interest regions from the block-based regions, we utilize the graph cuts method proposed by [33]. This method has been successfully applied to a wide range of problems in interactive object segmentation [24, 32, 99, 100, 101]. Suppose an image is represented by a graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$, where $\mathcal{V}$, $\mathcal{E}$ are defined as a set of all nodes (i.e., image pixels) and a set of all edges connecting neighbouring nodes, respectively. In this graph, there are two terminal nodes representing "ROI" and "background" labels (called source and sink nodes: $\{s, t\}$). Edges between pixels called n-links and an edge between a pixel and a terminal called t-link. In this framework, a segmentation energy function is formulated in terms of regional and boundary properties as

$$E(A) = \lambda \times R(A) + B(A) \qquad (4.11)$$

$$R(A) = \sum_{i \in \mathcal{V}} R_i(A_i) \tag{4.12}$$

$$B(A) = \sum_{\{i,j\} \in \mathcal{E}} B_{i,j} \times \delta_{A_i \neq A_j} \ , \delta_{A_i \neq A_j} = \begin{cases} 1 \text{ if } A_i \neq A_j \\ 0 \text{ if } A_i = A_j \end{cases} \tag{4.13}$$

where $A = \{A_i\}_{i=1}^{|\mathcal{V}|}$ denotes a binary vector (pixel class labels) whose elements A_i can be either "1" (i.e., ROI) or "0" (i.e., background). The coefficient $\lambda \geq 0$ controls the relative importance between $R(A)$ (i.e., regional term) and $B(A)$ (i.e., boundary term). $R(A)$ represents the costs of t-links where $B(A)$ represents the costs of n-links. The function $E(A)$ is optimized by solving the max-flow algorithm [95, 98]. In this framework, we considered the pixels of strongly sharp blocks as object seeds. In our ensemble clustering algorithm, these blocks have been labelled as sharp in both consecutive levels. The pixels of background blocks are also considered as background seeds. We adopted the boundary energy term as demonstrated in [100] using RGB colour information and constructed a graph model. The boundary term is computed by using the L2-Norm of the RGB colour difference of two pixels. We also used the max-flow algorithm to minimize the graph cut problem and as a result a corresponding segmentation achieved. Figure 4.12 and 4.13 shows a number of original low DOF images along with corresponding segmentation results by using graph cut modelling.

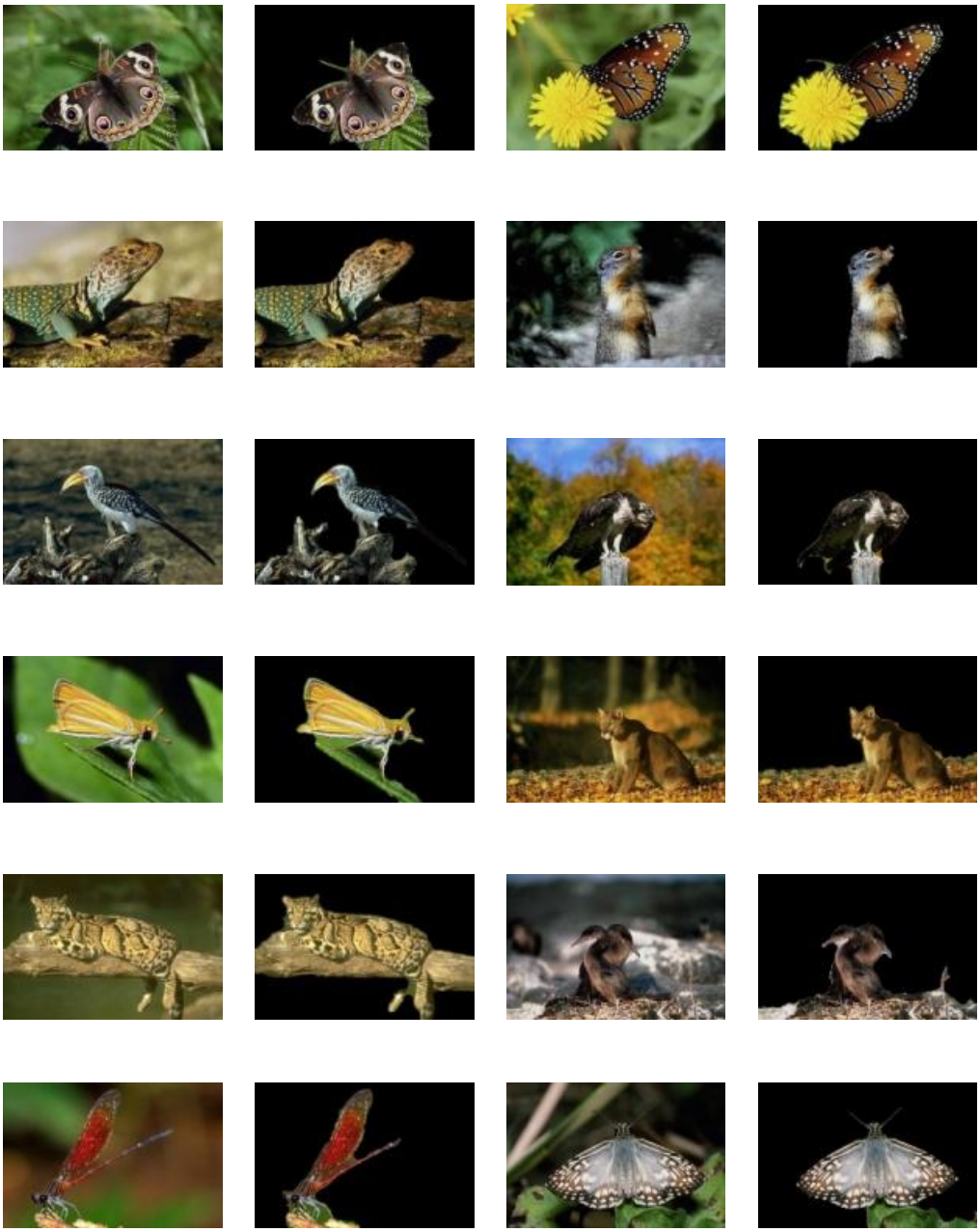


Figure 4.12: Original low DOF images (left) and corresponding segmentation results (right) obtained by the proposed approach.

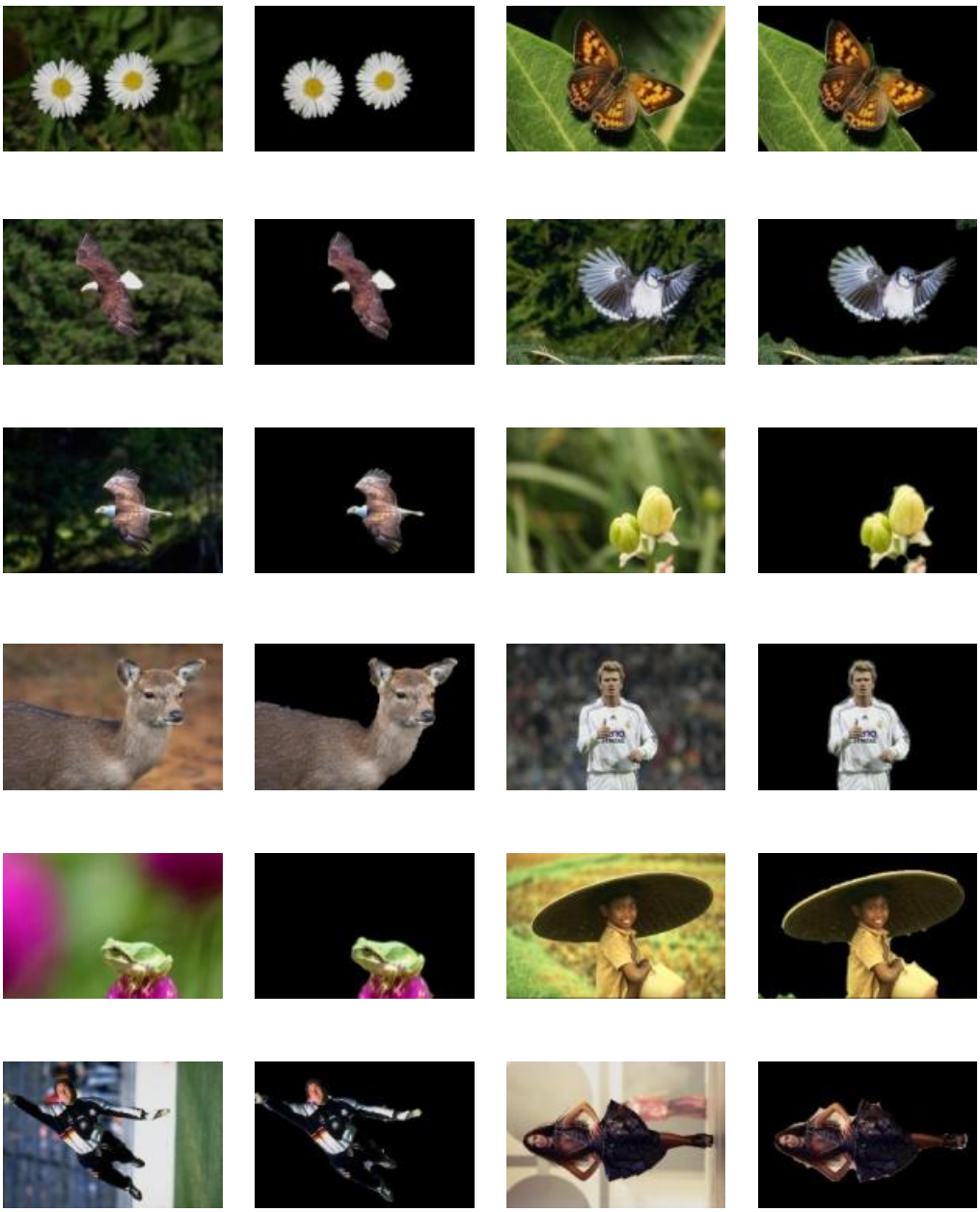


Figure 4.13: Original low DOF images (left) and corresponding segmentation results (right) obtained by the proposed approach.

4.4 Summary

In order to extract pixel-based interest regions from the block-based regions, two novel approaches presented. The first approach which employs grayscale component of the original image is based on optimising a threshold in a DOG image and morphological operations. This approach attempts to augment the visibility of low intensity variations of the regions and boundaries while reducing the effect of noise. The second approach is developed by incorporating our ensemble clustering approach and the colour-based graph cuts optimisation technique. Despite its simplicity and practical efficiency, it also provides a globally optimal solution for a binary segmentation problem.

Chapter 5

5. EXPERIMENTAL RESULTS AND COMPARISON

5.1 Introduction

In this Chapter, the accuracy, performance, and time efficiency of the proposed extraction approach is reported. Several experiments are conducted by using the precision-recall framework and two main datasets. To provide a strong comparison, our approach is also compared with the state-of-the-art supervised and unsupervised approaches. The generalisation ability of the proposed approach is also evaluated by using a specified range of image resolutions varying from 192×128 to 1536×1024. We also evaluate the influence of the propose ensemble EM clustering algorithm on the segmentation performance of the proposed approach.

5.2 Experimental Results

To analyze the performance of the proposed approach, several experiments have been carried out by using two main image datasets and adopting the precision-recall framework [102]. We selected more than 150 low DOF images of the size 384×256 or 256×384 from the Corel dataset [56-57, 70] including heterogeneous ROI and complex backgrounds. For this dataset, a number of segmentation results including block-based and pixel-based interest regions are provided. A sample of four test images of this dataset has been selected and our results are compared with the results of approaches in [40-44]. In addition, we used 117 Web images of different sizes between 260×180 and 400×374 and their ground-truth segmentations provided by [42] and compared our results with approaches in [40-44]. Moreover, the proposed approach is tested with a number of high resolution images selected from an online photo sharing website (www.Flickr.com). The approach has been tested on a Core2 Due 2.66GHz Intel processor and 2.00 GB of RAM using C++ and MATLAB version R2008a.

To provide numerical results and evaluate the performance of the proposed approach, the precision-recall framework is used [102-103]. We formulate low DOF image segmentation as a classification problem of discriminating ROI from the background and apply the precision-recall framework using manually segmented images from this dataset as ground-truths. The F-measure capturing a trade off as the weighted harmonic mean of precision and recall is defined as [102]

$$F = (1 + \alpha) \times Precision \times Recall / (\alpha \times Recall + Precision) \qquad (5.1)$$

where $Precision = TP/(TP + FP)$, $Recall = TP/(TP + FN)$. $TP, FP,$ and FN denote the number of true positive, false positive and false negative pixels, respectively. Positive and negative terms are denoted as ROI and background in (5.1). In this evaluation, the average values of precision, recall, and F-measure over test images are computed. We also use $\alpha = 0.3$, $\alpha = 0.5$, and $\alpha = 0.7$ in our experiments according to [44,104].

5.2.1 Corel Dataset Images

We selected a sample of four test images from the Corel dataset, namely *football*, *butterfly*, *leopard*, and *bird*. These images have been used as a benchmark by several researchers in this field [3, 40-41]. As shown in Figure 5.1, all approaches including the proposed approach achieve satisfactory visual segmentation results for the *football*, *butterfly*, and *bird* images. However, for the *leopard* image, the proposed approach achieves a better visual segmentation result than the other approaches when compared with the manually segmented binary image shown in column (b). The main reason for this improvement is that noisy regions including high-frequency components are removed from the image background during the first stage of the proposed approach. For the segmentation result of [40], high frequency components in both foreground and background have been removed which results the lowest segmentation performance (i.e., F-measure). In [41-44], the approaches extracted the ROI as well as a part of the unwanted high-frequency regions (i.e., error) in the background. In Figure 5.2, these errors are denoted as false negative and false positive for the segmentation results of [40] and [44], respectively. Table 5.1 demonstrates a comparison of average segmentation performance for the four

test images of Corel dataset. As evident from Table 5.1, it is observed that our proposed approach presents better performance than other approaches for all the four test images. Numerical results are calculated by the F-measure criterion presented in (5.1). Figure 5.3 shows some examples of 150 final segmentation results from Corel dataset obtained from our approach.

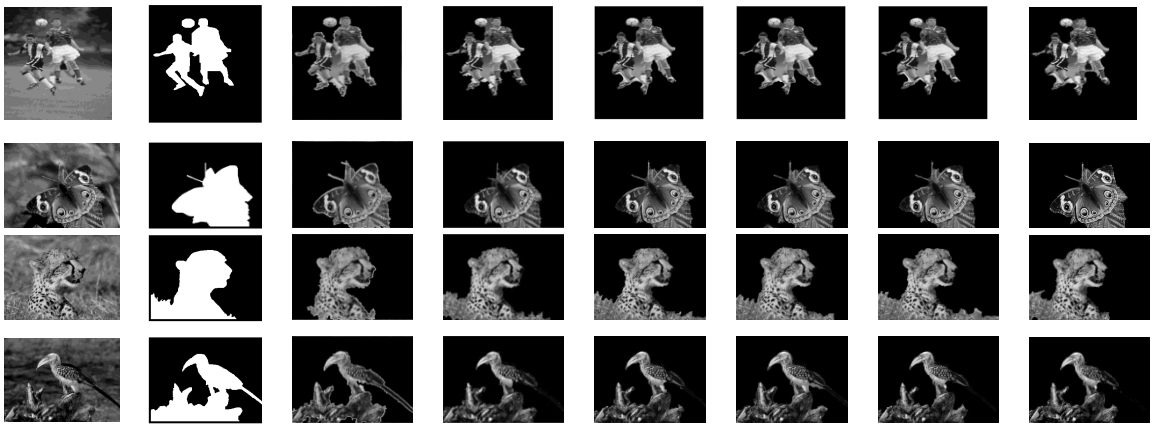


Figure 5.1: Visual comparison of segmentation results for the Corel dataset images, namely *football*, *butterfly*, *leopard*, and *bird* from top to bottom, respectively. (a) Low DOF images. (b) References by human segmentation [41]. (c)-(g) Results from [40-44], respectively. (h) Results from our approach.

(a) (b)

Figure 5.2: Illustration of the error in the background (false negative) and foreground (false positive) regions obtained from [40] (a) and [44] (b), respectively.

Table 5.1

Comparison of average F-measure, precision, and recall for the four test images selected from Corel dataset.

Approaches	**F-measure (%)** $\alpha = 0.3$	**F-measure (%)** $\alpha = 0.5$	**F-measure (%)** $\alpha = 0.7$	**Precision (%)**	**Recall (%)**
[40]	84.36	83.50	82.85	86.37	78.30
[41]	85.06	84.39	83.88	86.62	80.26
[42]	85.92	85.43	85.05	87.04	82.37
[43]	85.67	85.09	84.66	86.99	81.54
[44]	85.01	84.31	83.79	86.61	80.06
Proposed	**86.72**	**86.15**	**85.72**	**88.03**	**82.26**

Figure 5.3: Segmentation results for gray-level low DOF images selected from the Corel dataset.

5.2.2 117 Web Images

To provide further evaluation and comparison, our approach is also tested on 117 test images, which have been provided by [42]. Figure 5.4 illustrates a number of segmentation results obtained from our proposed approach as well as the different approaches [40-44] for this dataset.

Figure 5.4: Segmentation results for the test images provided by [42]. (a) Grayscale component of the original low DOF images. (b) Manually segmented binary images provided by [42] (i.e., ground-truth masks).). (c), (d), (e), (f), and (g) Results obtained from approaches [40-44], respectively. (h) Results from our unsupervised approach.

Table 5.2 shows the comparison of average F-measure, precision, and recall values for different segmentation methods. As evident from Table 2, the proposed method achieves the highest segmentation performance (F-measure) compared with the state-of-the-art approaches.

Table 5.2

Comparison of average F-measure, precision, and recall values for the 117 test images.

Approaches	F-measure (%) $\alpha = 0.3$	F-measure (%) $\alpha = 0.5$	F-measure (%) $\alpha = 0.7$	Precision (%)	Recall (%)
[40]	84.26	81.60	79.68	90.93	67.71
[41]	84.77	83.37	82.33	88.10	75.28
[42]	89.41	88.54	87.90	91.41	83.32
[43]	88.71	87.83	87.16	90.78	82.46
[44]	85.80	83.60	81.99	91.20	71.66
Proposed	**91.31**	**90.20**	**89.37**	**93.91**	**83.60**

Table 5.3 compares the average computational time between the proposed approach and the existing approaches for the 117 test images.

Table 5.3

Comparison of average computational time results for the 117 test images.

Approaches	Learning	Average Computational Time (Second)
[40]	Unsupervised	7.9967
[41]	Unsupervised	> 8.0
[42]	Supervised	4.635
[43]	Unsupervised	> 8.0
[44]	Unsupervised	7.0
Proposed	**Unsupervised**	**2.2836**

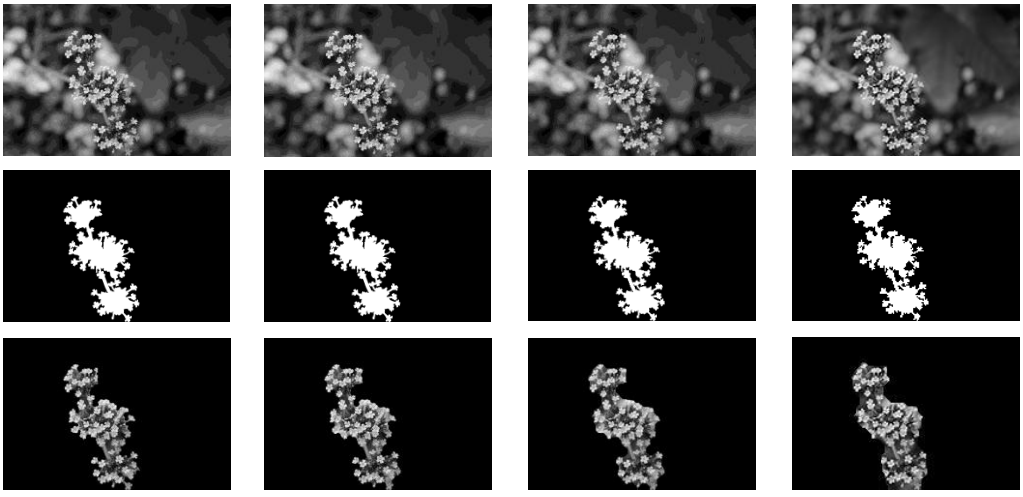
Because of space limitation in this thesis, a specified number of segmentation results have been illustrated in Figures 5.5-5.7. Segmentation results for more than 250 low DOF images may be seen at http://www.labvision.co.uk.

Figure 5.5: A number of segmentation results for gray-level low DOF images (left: original image, right: segmentation result)

Figure 5.6: A number of segmentation results for gray-level low DOF images (left: original image, right: segmentation result)

Figure 5.7: A number of segmentation results for gray-level low DOF images (left: original image, right: segmentation result)

To further demonstrate the performance and generalization ability of the proposed approach, we tested it over a specified range of resolutions varying from 192×128 to 1536×1024 for 50 images selected from an online photo sharing website [105] using a fixed set of parameter values. Figure 5.8 illustrates a typical example of the selected images in different resolutions along with their visual segmentation results obtained from our approach. In Table 5.4, numerical results including average F-measure and computational time as well as parameter values for the images have been presented. From the obtained results, it is observed that the higher resolution of an image provides the higher segmentation performance. However, the more computational time will be required to extract an ROI from a high resolution image.



a) 1536×1024 b) 768×512 c) 384×256 d) 192×128

Figure 5.8: ROI extraction results in different resolutions for an image. Original grayscale image (first row), manually segmented binary image (second row), and corresponding ROI result (third row) obtained from the proposed approach in four different resolutions (a)-(d).

Table 5.4

Illustration of parameter values (parent and child block size and ω), average computational time and F-measure in the specified resolutions of images. The following results are based on 50 images for each resolution.

Image Resolution	a) 1536×1024	b) 768×512	c) 384×256	d) 192×128
Parent and Child Block Size	256×256, 128×128	128×128, 64×64	64×64, 32×32	32×32, 16×16
$\boldsymbol{\omega}$	128	64	32	16
Average Computational Time (Sec.)	7.2	2.63	2.18	1.83
Average F-measure (%) $\boldsymbol{\alpha = 0.3}$	94.23	93.99	91.68	87.50

5.2.3 Average F-measure over the 117 Images for Different Values of z

The 117 test images are chosen to assess the effect of changing threshold z on the performance of the approach. We selected a set of discrete neighborhood thresholds of z centered at z^*, where $z^* \in [z_1, z_2]$ is the threshold obtained from (4.7) described in Section 4.2.2. Suppose for every image i, the thresholds and the F-measure values are represented by $z_i = \left(z_{i,-j}, z_{i,-j+0.5}, \dots, z_{i,j}\right)$ and $\mathcal{F}_i = \left(\mathcal{F}_{i,-j}, \mathcal{F}_{i,-j+0.5}, \dots, \mathcal{F}_{i,j}\right)$, respectively, where $i = 1, \dots, M$, M is the total number of images, $j = 3$, and $z_{i,0} = z_i^*$. The F-measure of image i corresponding to the threshold z_i^* is also denoted by $\mathcal{F}_{i,0}$. In all experiments, we found that considering 6 neighbors greater and 6 neighbors less than the value z_i^* is good enough for the evaluation. The average F-measure

value for a set of thresholds labelled by $k \in \{-3, -2.5, \ldots, 2.5, 3\}$ is computed by $\bar{\mathcal{F}}_k = \frac{1}{M}\sum_{i=1}^{M} \mathcal{F}_{i,k}$. Fig. 5.9 illustrates the relationship between a set of thresholds and the average F-measure values of the approach for the 117 images. As the result shows, the average F-measure for the threshold $z_{i,0}$, which is obtained from (4.7), is the maximum among a set of neighborhood thresholds.

Figure 5.9: Average F-measure values versus a set of thresholds for the 117 test images.

5.2.4 Segmentation Performance without using Ensemble EM Clustering Algorithm

We evaluated the influence of the proposed ensemble EM clustering algorithm on the segmentation performance of the approach. We tested our approach on the 117 test images using the EM clustering instead of the ensemble EM clustering algorithm. In this experiment, the EM clustering algorithm [74] is independently utilized at two consecutive levels of block size and consequently two partitions obtained. The obtained partitions are then combined according to Section 3.2 to create a clustered image. The second stage of the approach remains unchanged in this experiment. We also set $\alpha = 0.3$ and

computed the average precision, recall, and F-measure. Figure 5.10 shows a comparison of average segmentation performance of the approach with and without the ensemble EM clustering algorithm. This comparison shows the advantage of using the ensemble EM clustering algorithm. This reduction in average segmentation performance with the EM clustering algorithm originates from the fact that the EM as a local method is dependent on its initialization process and cannot necessarily guarantee to find the best partition (i.e., global optimum).

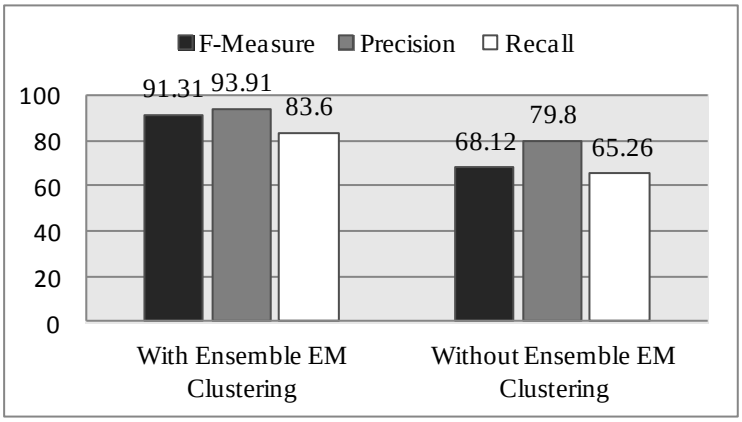


Figure 5.10: Comparison of average segmentation performance (F-measure (%), Precision, and Recall) when using the ensemble EM clustering algorithm and without the ensemble EM clustering on the 117 test images.

5.2.5 Evaluation of the Combining Process

To quantitatively evaluate the reliability of combining the blocks of the two consecutive levels, we randomly chose 50 low DOF images from the 117 test images. In this experiment, we firstly created 50 ground-truth images at the level of block size. All blocks of the ground-truth images at level two have been manually labelled based on the definition of the three region classes outlined in

Section 3.4. Then, we ran the ensemble EM clustering algorithm on the 50 selected test images which resulted in 50 final partitions. We considered the situation in which the cluster label of a parent block is sharp or uncertain. In this evaluation, we achieved 98% accuracy for combining the blocks of two consecutive levels. The 2% error occurs when the cluster label of a parent block is uncertain and its child block is sharp. The obtained result from the combination of the uncertain parent and its sharp child resulted in sharp label, whereas the manually labelled block indicates uncertain label. However, this error can be neglected as the blocks with sharp and uncertain cluster labels are integrated and considered as block-based interest regions.

5.2.6 Segmentation Performance using Graph Cut Modelling

Figure 5.11 shows the comparison of average F-measure, precision, and recall values for different unsupervised segmentation methods. As evident from Figure 5.11, the proposed method with 91.7% average F-measure value outperforms existing unsupervised approaches for extracting the ROI in low DOF images, which shows an improvement of 5.9%. In addition, identifying significant regions at the level of the block size and utilizing the minimal graph cut segmentation algorithm makes our approach more time efficient compared to the methods that employ whole image pixels for the segmentation process. Table 5.5 compares the average computational time between the proposed approach and the existing methods for the 117 test images on a same platform. Approximately 50% reduction of the average computational time by using the proposed approach is evident for unsupervised approach [44].

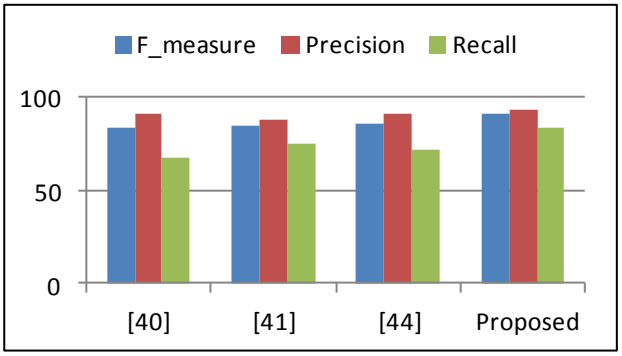


Figure 5.11: Segmentation performance comparison between the proposed approach using graph cut modelling and the state-of-the-art approaches.

Table 5.5

Comparison of average computational time results for unsupervised learning approaches over the 117 test images.

Approaches	Average Computational Time (Second)
[40]	7.9967
[41]	> 8.0
[42]	> 8.0
[43]	7.0
Proposed	**3.4825**

5.3 Discussion and Summary

In this Chapter, we tested our proposed approach with different low DOF images selected from two datasets. The approach has also been compared with the existing unsupervised and supervised approaches in [40-44]. As evident from Table 5.2, our approach with 91.3% average F-measure value outperforms existing state-of-the-art approaches for extracting the ROI in low DOF images. For the best unsupervised approach [43] compared with our proposed approach the improvement is 2.6%. Similarly, for the supervised approach [42] the improvement is 1.9%. In addition, identifying significant regions at the level of the block size and utilizing morphological operations make our approach more time efficient compared with the methods that employ whole image pixels for the segmentation process. As illustrated by Table 5.3, approximately 33% and 50% reductions of the average computational time by using the proposed approach are evident for unsupervised [44] and supervised [42] approaches, respectively. The reduction in computational time is caused by two main reasons: 1) relying on the texture (i.e., energy and contrast) details of the regions rather than colour information, 2) identifying salient regions in the block-wise ensemble clustering process. This demonstrates that our approach while running on a slower platform is computationally more efficient than the other approaches. This major advantage would be applicable to a number of region-based image retrieval applications that require online processing such as image/video target searching and indexing.

Chapter 6

6. CONCLUSION AND FUTURE WORK

6.1 Conclusion

The problem of efficient and effective interest region (i.e., focused regions) extraction from still images is of great practical importance in computer vision applications. Extracting meaningful and relevant regions in an image is a major step toward image understanding and still remains an open problem. The capability of extracting focused regions can help to bridge the semantic gap by integrating image regions which are relevant and generally do not exhibit uniform visual characteristics. There have been several unsupervised and supervised approaches to extract important objects from a complex background

in a low DOF image. However, as discussed in this thesis, the existing approaches are not successful due to two main reasons: 1) dependency on high frequency components, 2) high computational complexity. Exploiting high frequency components alone often results in errors in both ROI and background regions. In background regions, despite blurring, there could be noisy regions (i.e., busy-texture or high contrast) in which high frequency components are still strong enough. Therefore, these regions may be classified mistakenly as focused regions. On the other hand, focused regions including constant gray-level values are prone to be misclassified as defocused regions. Therefore, identifying ROI regions using high frequency components should be incorporated with other cues or some supplementary techniques. High computational complexity is another challenging aspect of ROI extraction task. To extract focused regions, most existing methods employ the whole pixels of a low DOF image which makes the method too complex and consequently inefficient.

Block-based ROI extraction technique proposed in this thesis is a reliable solution addressing existing problems. As discussed in Chapter 3, this technique utilises a two-level block-based clustering process. In this sense, we faced with a common problem of local optima experienced in clustering algorithms. Optimisation-based clustering algorithms such as k-means and EM clustering are highly dependent on initialisation process. Therefore, an ensemble technique is developed and incorporated into the EM clustering algorithm which improves the robustness and the stability of the clustering process. As our technique utilise two levels of resolution, a fusion decision approach is also presented in which the blocks of two consecutive levels are appropriately combined. The

reliability of this combination has been demonstrated in Chapter 5. The final clustering result obtained from the proposed ensemble EM clustering approach includes the ROI and background at the level of block size. As segmentation results demonstrate, the background regions even including high frequency components are successfully recognised and separated in this stage. Therefore, this stage overcomes the weaknesses of the current approaches which rely only on high frequency components.

To extract the ROI regions at the level of pixels, two approaches are presented. In the first approach, which is one of the main parts of our approach, a novel methodology is presented by determining optimum threshold and using morphological operations. In this methodology, which employs the grayscale component of the original image, a threshold in a DOG image is optimised and consequently a binary RSM of all smooth and focused regions from the block-based ROI is created. To identify the underlying regions shape and the boundary of an object(s) in obtained RSM, a set of morphological operations is appropriately employed. Visual segmentation results illustrate the effectiveness of the proposed methodology. The second approach presented in Chapter 4 aims to extract interest regions at the level of pixels by using a colour-based graph cut modelling. In this approach, a minimal graph cut is constructed using object and background seeds provided by the ensemble EM clustering algorithm.

Several experiments have been carried out to illustrate the performance and time efficiency of the proposed approach. Low DOF image segmentation can be considered as a classification problem of discriminating ROI form the background and therefore the precision-recall framework is used. Two main

image datasets including a specified range of busy-texture (i.e., noisy) and smooth regions have been employed to test the proposed approach. As discussed in Chapter 5, our main approach (i.e., the ensemble EM clustering along with determining a threshold) with 91.3% average F-measure value outperforms existing state-of-the-art approaches for extracting the ROI in low DOF images. For the best unsupervised approach [43] compared with our proposed approach the improvement is 2.6%. Similarly, for the supervised approach [42] the improvement is 1.9%. In terms of time efficiency, approximately 33% and 50% reductions of the average computational time by using the proposed approach are evident for unsupervised [44] and supervised [42] approaches, respectively. This demonstrates that our approach while running on a slower platform is computationally more efficient than the other approaches. This major advantage would be applicable to a number of region-based image retrieval applications that require online processing such as image/video target searching and indexing. The second approach (i.e., the ensemble EM clustering along with graph cut modelling) with 91.7% average F-measure value outperforms existing unsupervised approaches for extracting the ROI in low DOF images, which shows an improvement of 5.9%. Approximately 50% reduction of the average computational time by using the proposed approach is evident for unsupervised approach [44].

6.2 Recommendation for Future Research

Based on the research presented in this thesis, other possible investigation and exploration research can be initiated. This ROI extraction research can be extended by several ideas.

- The proposed approach in this thesis has confirmed the possibility of online processing. However, this possibility should be further examined by a practical evaluation.
- The proposed approach does not utilise any pixel refinement method for extracting the object boundary. Therefore, a colour based refinement approach may help to more accurately extract the boundary of ROI regions.
- In the proposed research, we accessed to only a dataset of 117 low DOF images along with their ground-truth segmentation masks. The rest of test images used in this thesis which have been selected from Corel dataset and Flicker, an online photo sharing website [105], are without ground-truth segmentation masks. Therefore, a new dataset of a large number of low DOF images and their corresponding ground-truth segmentation masks need to be provided.
- In the current TV programs and film productions, low DOF has become an important technique to highlight the main objects in order to attract user attention in this scene. Therefore, it would be desirable to test and evaluate the performance of the proposed approach using video frames.

Chapter 7

7. REFERENCES

[1] A. Adam, *The Camera*: Boston: New York Graphic Society, 1980.

[2] S. Kuthirummal, H. Nagahara, Z. Changyin, and S. K. Nayar, "Flexible Depth of Field Photography," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 33, pp. 58-71, 2011.

[3] J. Z. Wang, L. Jia, R. M. Gray, and G. Wiederhold, "Unsupervised multiresolution segmentation for images with low depth of field," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 23, pp. 85-90, 2001.

[4] D.-M. Tsai and H.-J. Wang, "Segmenting focused objects in complex visual images," *Pattern Recognition Letters,* vol. 19, pp. 929-940, 1998.

[5] Y. Boykov and D. P. Huttenlocher, "A new Bayesian framework for object recognition," in *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 1999, p. 523 vol. 2.

[6] L. Kovacs and T. Sziranyi, "Image Indexing by Focus Map," in *Proc. Seventh Int'l Conf. Advanced Concepts for Intelligent Vision Systems*. vol. 3708, 2005, pp. 300-307.

[7] G. Rafiee, S. S. Dlay, and W. L. Woo, "A review of content-based image retrieval," in *Communication Systems Networks and Digital Signal Processing (CSNDSP), 2010 7th International Symposium on*, pp. 775-779.

[8] A. Smeulders, W. Worring, S. Santini, A. Gupta, and R. Jain, "Content-Based Image Retrieval at the End of the Early Years," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 22, pp. 1349-1380, 2000.

[9] R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *ACM Computing Surveys,* vol. 40, pp. 1-60, 2008.

[10] Y. Liu, D. Zhang, G. Lu, and W. Ma, "A survey of content-based image retrieval with high-level semantics," *The Journal of the Pattern Recognition Society,* vol. 40, pp. 262-282, 2007.

[11] D. Hoiem, R. Sukthankar, H. Schneiderman, and L. Huston, "Object-based image retrieval using the statistical structure of images," in *Computer Vision and Pattern Recognition, ,* 2004, pp. II-490-II-497 Vol.2.

[12] C. Guo and L. Zhang, "A Novel Multiresolution Spatiotemporal Saliency Detection Model and Its Applications in Image and Video Compression," *Image Processing, IEEE Transactions on,* vol. 19, pp. 185-198, 2010.

[13] G. Rafiee, S. S. Dlay, and W. L. Woo, "Automatic Segmentation of Interest Regions in Low Depth of Field Images Using Ensemble Clustering and Graph Cut Optimization Approaches," in *Multimedia (ISM), 2012 IEEE International Symposium on*, pp. 161-164.

[14] A. Cavallaro, O. Steiger, and T. Ebrahimi, "Tracking video objects in cluttered background," *Circuits and Systems for Video Technology, IEEE Transactions on,* vol. 15, pp. 575-584, 2005.

[15] T. Uchiyama and M. A. Arbib, "Color image segmentation using competitive learning," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 16, pp. 1197-1206, 1994.

[16] M. Mirmehdi and M. Petrou, "Segmentation of color textures," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 22, pp. 142-159, 2000.

[17] J. Shi and J. Malik, "Normalized cuts and image segmentation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 22, pp. 888-905, 2000.

[18] Y. Deng and B. S. Manjunath, "Unsupervised segmentation of color-texture regions in images and video," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 23, pp. 800-810, 2001.

[19] C. Carson, S. Belongie, H. Greenspan, and J. Malik, "Blobworld: image segmentation using expectation-maximization and its application to

image querying," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 24, pp. 1026-1038, 2002.

[20] D. Guo and X. Ming, "Color clustering and learning for image segmentation based on neural networks," *Neural Networks, IEEE Transactions on,* vol. 16, pp. 925-936, 2005.

[21] W. Tao, H. Jin, and Y. Zhang, "Color Image Segmentation Based on Mean Shift and Normalized Cuts," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on,* vol. 37, pp. 1382-1389, 2007.

[22] Q. Yu and D. A. Clausi, "IRGS: Image Segmentation Using Edge Penalties and Region Growing," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 30, pp. 2126-2139, 2008.

[23] L. Garcia Ugarriza, E. Saber, S. R. Vantaram, V. Amuso, M. Shaw, and R. Bhaskar, "Automatic Image Segmentation by Dynamic Region Growth and Multiresolution Merging," *Image Processing, IEEE Transactions on,* vol. 18, pp. 2275-2288, 2009.

[24] B. Peng, L. Zhang, D. Zhang, and J. Yang, "Image segmentation by iterated region merging with localized graph cuts," *Pattern Recognition,* vol. 44, pp. 2527-2538, 2011.

[25] C. S. Won, K. Pyun, and R. M. Gray, "Automatic object segmentation in images with low depth of field," in *Image Processing, International Conference on*, 2002, pp. 805-808 vol.3.

[26] J. R. Smith and S.-F. Chang, "Transform features for texture classification and discrimination in large image databases," in *Image Processing, IEEE International Conference*, 1994, pp. 407-411 vol.3.

[27] J. R. Smith and S.-F. Chang, "Local color and texture extraction and spatial query," in *Image Processing, IEEE International Conference on*, 1996, pp. 1011-1014 vol.3.

[28] J. R. Smith and S.-F. Chang, "Visually searching the Web for content," *Multimedia, IEEE Transactions on*, vol. 4, pp. 12-20, 1997.

[29] F. Jing, M. Li, H.-J. Zhang, and B. Zhang, "Unsupervised image segmentation using local homogeneity analysis," in *Circuits and Systems, International Symposium on*, 2003, pp. II-456-II-459 vol.2.

[30] S. X. Yu and J. Shi, "Segmentation given partial grouping constraints," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, pp. 173-183, 2004.

[31] L. Grady, "Random Walks for Image Segmentation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 28, pp. 1768-1783, 2006.

[32] C. Rother, V. Kolmogorov, and A. Blake, ""GrabCut": interactive foreground extraction using iterated graph cuts," in *ACM SIGGRAPH* Los Angeles, California: ACM, 2004.

[33] Y. Y. Boykov and M. P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images," in *Computer Vision, ICCV, Eighth IEEE International Conference on*, 2001, pp. 105-112 vol.1.

[34] H. Lombaert, Y. Sun, L. Grady, and C. Xu, "A multilevel banded graph cuts method for fast image segmentation," in *Computer Vision, ICCV, Tenth IEEE International Conference on*, 2005, pp. 259-265 Vol. 1.

[35] S. Vicente, V. Kolmogorov, and C. Rother, "Graph cut based image segmentation with connectivity priors," in *Computer Vision and Pattern Recognition, CVPR, IEEE Conference on*, 2008, pp. 1-8.

[36] Z. Liu, W. Li, L. Shen, Z. Han, and Z. Zhang, "Automatic segmentation of focused objects from images with low depth of field," *Pattern Recognition Letters*, vol. 31, pp. 572-581, 2010.

[37] Z. Ye and C.-C. Lu, "Unsupervised multiscale focused objects detection using hidden Markov tree," in *Computer Vision, Pattern Recognition, and Image Processing, International Conference on*, 2002, pp. 1-4.

[38] H. Tong, L. Mingjing, Z. Hongjiang, and Z. Changshui, "Blur detection for digital images using wavelet transform," in *Multimedia and Expo, IEEE International Conference on*, 2004, pp. 17-20 Vol.1.

[39] N. Neverova and H. Konik, "Edge-based method for sharp region extraction from low depth of field images," in *Visual Communications and Image Processing (VCIP), IEEE Confterence on*, 2008, pp. 1-6.

[40] C. Kim, "Segmenting a low-depth-of-field image using morphological filters and region merging," *Image Processing, IEEE Transactions on*, vol. 14, pp. 1503-1511, 2005.

[41] H. Li and K. N. Ngan, "Unsupervized Video Segmentation With Low Depth of Field," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 17, pp. 1742-1751, 2007.

[42] H. Li and K. N. Ngan, "Learning to Extract Focused Objects from Low DOF Images," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 21, pp. 1571-1580, 2011.

[43] F. Graf, H. P. Kriegel, and M. Weiler, "Robust segmentation of relevant regions in low depth of field images," in *Image Processing, IEEE International Conference on*, 2011, pp. 2861-2864.

[44] T. Chen and H. Li, "Segmenting focused objects based on the Amplitude Decomposition Model," *Pattern Recognition Letters*, vol. 33, pp. 1536-1542, 2012.

[45] L. Kovacs and T. Sziranyi, "Focus Area Extraction by Blind Deconvolution for Defining Regions of Interest," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, pp. 1080-1085, 2007.

[46] Rajashekhara and C. Subhasis, "Segmentation and region of interest based image retrieval in low depth of field observations," *Image Vision Computing*, vol. 25, pp. 1709-1724, 2007.

[47] C. Zhang and H. Zhang, "An unsupervised approach to determination of main subject regions in images with low depth of field," in *Multimedia Signal Processing, IEEE 10th Workshop on*, 2008, pp. 650-653.

[48] H. Li and K. N. Ngan, "Unsupervised Segmentation of Defocused Video Based on Matting Model," in *Image Processing, IEEE International Conference on*, 2006, pp. 1825-1828.

[49] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*: Prentice Hall, 2002.

[50] D. G. Lowe, "Object recognition from local scale-invariant features," in *Computer Vision, The Seventh IEEE International Conference on*, 1999, pp. 1150-1157 vol.2.

[51] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, pp. 91-110, 2004.

[52] T. Lindeberg, "Scale-space theory: a basic tool for analyzing structures at different scales," *Journal of Applied Statistics*, vol. 21, pp. 225-270, 1994.

[53] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 9, pp. 62-66, 1979.

[54] O. D. Trier and A. K. Jain, "Goal-directed evaluation of binarization methods," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 17, pp. 1191-1201, 1995.

[55] R. Liu, Z. Li, and J. Jia, "Image partial blur detection and classification," in *Computer Vision and Pattern Recognition, IEEE Conference on*, 2008, pp.1-8.

[56] J. Z. Wang, L. Jia, and G. Wiederhold, "SIMPLIcity: semantics-sensitive integrated matching for picture libraries," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 23, pp. 947-963, 2001.

[57] J. Li and J. Z. Wang, "Automatic Linguistic Indexing of Pictures by a statistical modeling approach," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 25, pp. 1075-1088, 2003.

[58] S. G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 11, pp. 674-693, 1989.

[59] I. Daubechies, "The wavelet transform, time-frequency localization and signal analysis," *Information Theory, IEEE Transactions on*, vol. 36, pp. 961-1005, 1990.

[60] I. Daubechies, *Ten Lectures on Wavelets*: Capital City Press, 1992.1005, 1990.

[61] P. S. Addison, *The Illustrated Wavelet Transform Handbook*: Taylor & Francis Group, 2002.

[62] P. Porwik and A. Lisowska, "The haar wavelet transform in digital image processing: Its status and achievements," *Machine Graphics and Vision*, vol. 13, pp. 79–98, 2004.

[63] G. Gan, C. Ma, and J. Wu, *Data Clustering: Theory, Algorithms, and Applications*: Society for Industrial and Applied Mathematics, 2007.

[64] S. Theodoridis and K. Koutroumbas *Pattern Recognition*, Third ed.: Academic Press, 2006.

[65] A. K. Jain, "Data clustering: 50 years beyond K-means," *Pattern Recognition Letters*, vol. 31, pp. 651-666, 2010.

[66] 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 Workshop on Generative-Model Based Vision*, 2004.

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

[68] M. Everingham, M. a. L. Van~Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes (VOC) Challenge," *International Journal on Computer Vision, Springer,* 2009.

[69] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Computer Vision, ICCV Eighth IEEE International Conference on*, 2001, pp. 416-423 vol.2.

[70] J. Tang and P. H. Lewis, "A Study of Quality Issues for Image Auto-Annotation with the Corel Dataset," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 17, pp. 384-389, 2007.

[71] J. Z. Wang, D. Geman, J. Luo, and R. M. Gray, "Real-World Image Annotation and Retrieval: An Introduction to the Special Section," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 30, pp. 1873-1876, 2008.

[72] J. Li and J. Z. Wang, "Real-Time Computerized Annotation of Pictures," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 30, pp. 985-1002, 2008.

[73] M. Unser, "Texture classification and segmentation using wavelet frames," *Image Processing, IEEE Transactions on,* vol. 4, pp. 1549-1560, 1995.

[74] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," Journal of the Royal Statistical Society, vol. 34B, pp. 1-38, 1977.

[75] M. A. T. Figueiredo and A. K. Jain, "Unsupervised learning of finite mixture models," Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 24, pp. 381-396, 2002.

[76] D. Manning and H. Schütze, Foundations of Statistical Natural Language Processing Cambridge, MA: The MIT Press, 1999.

[77] A. L. N. Fred and A. K. Jain, "Data clustering using evidence accumulation," in Pattern Recognition, 16th International Conference on, 2002, pp. 276-280 vol.4.

[78] A. L. N. Fred and A. K. Jain, "Robust data clustering," in Computer Vision and Pattern Recognition, IEEE Computer Society Conference on, 2003, pp. 128-33 vol.2.

[79] C. Ji and S. Ma, "Combinations of weak classifiers," Neural Networks, IEEE Transactions on, vol. 8, pp. 32-42, 1997.

[80] L. I. Kuncheva, Combining Pattern Classifiers, Methods and Algorithms: John Wiley and Sons, 2004.

[81] A. Strehl and J. Ghosh, "Cluster Ensembles - A Knowledge Reuse Framework for Combining multiple partitions," Journal of Machine Learning Research vol. 3, pp. 583-617, 2003.

[82] T. Lange and M. B. Joachim, "Combining partitions by probabilistic label aggregation," in Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining Chicago, Illinois, USA: ACM, 2005.

[83] A. Topchy, A. K. Jain, and W. Punch, "Clustering ensembles: models of consensus and weak partitions," Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 27, pp. 1866-1881, 2005.

[84] J. Azimi, M. Mohammadi, A. Movaghar, and M. Analoui, "Clustering Ensembles Using Genetic Algorithm," in Computer Architecture for Machine Perception and Sensing, International Workshop on, 2007, pp. 119-123.

[85] I. T. Christou, "Coordination of Cluster Ensembles via Exact Methods," Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 33, pp. 279-293, 2011.

[86] J. Malik, S. Belongie, J. Shi, and T. Leung, "Textons, contours and regions: cue integration in image segmentation," in *Computer Vision, The Proceedings of the Seventh IEEE International Conference on*, 1999, pp. 918-925 vol.2.

[87] J. Malik, S. Belongie, T. Leung, and J. Shi, "Contour and texture analysis for image segmentation.," International Journal of Computer Vision, vol. 43.1, pp. 7-27, 2001.

[88] D. G. Lowe, "Object recognition from local scale-invariant features," in Computer Vision, The Proceedings of the Seventh IEEE International Conference on, 1999, pp. 1150-1157 vol.2.

[89] W. Zhuozheng, J. Kebin, and L. Pengyu, "A Novel Image Retrieval Algorithm Based on ROI by Using SIFT Feature Matching," in MultiMedia and Information Technology, International Conference on, 2008, pp. 338-341.

[90] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in Computer Vision and Pattern Recognition, IEEE Conference on, 2009, pp. 1597-1604.

[91] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," International journal of computer vision, vol. 60.2, pp. 91-110, 2004.

[92] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," in Computer Vision and Pattern Recognition, IEEE Computer Society Conference on, 2003, pp. II-257-II-263 vol.2.

[93] W. K. Pratt, Digital Image Processing: John Wiley & Sons, Inc., 4th edition, 2007.

[94] M. Petrou and P. Costas, Image processing: the fundamentals, Second ed.: Wiley, 2010.

[95] D. M. Greig, B. T. Porteous, and A. H. Seheult., "Exact maximum a posteriori estimation for binary images," Journal of the Royal Statistical Society, pp. 271-279, 1989.

[96] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," International Journal of Computer Vision, vol. 59.2, pp. 167-181, 2004.

[97] Y. Boykov and G. Funka-Lea, "Graph cuts and efficient ND image segmentation," International Journal of Computer Vision, vol. 70.2, pp. 109-131, 2006.

[98] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max- flow algorithms for energy minimization in vision," Pattern

Analysis and Machine Intelligence, IEEE Transactions on, vol. 26, pp. 1124-1137, 2004.

[99] C. Jung, B. Kim, and C. Kim, "Automatic segmentation of salient objects using iterative reversible graph cut," in Multimedia and Expo (ICME), IEEE International Conference on, pp. 590-595.

[100] Y. Fu, J. Cheng, Z. Li, and H. Lu, "Saliency Cuts: An automatic approach to object segmentation," in Pattern Recognition, ICPR, 19th International Conference on, 2008, pp. 1-4.

[101] C. Jung and C. Kim, "A Unified Spectral-Domain Approach for Saliency Detection and Its Application to Automatic Object Segmentation," Image Processing, IEEE Transactions on, vol. 21, pp. 1272-1283.

[102] C. Van Rijsbergen, *Information Retrieval*, second ed.: Dept. of Computer Science, University of Glasgow, 1979.

[103] X. Ren and J. Malik, "Learning a classification model for segmentation," in Computer Vision, Ninth IEEE International Conference on, 2003, pp. 10-17 vol.1.

[104] D. R. Martin, C. C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 26, pp. 530-549, 2004.

[105] "Flickr photo management and sharing application [online]," Available: http://www.flikr.com/, 2009.