University of Alberta

MACHINE LEARNING FOR ADAPTIVE PARAMETER SELECTION IN IMAGE SEGMENTATION

by

Xiaoli Wang 

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of **Master of Science**.

Department of Computing Science

Edmonton, Alberta
Fall 2006

Your file Votre référence
ISBN: 978-0-494-22403-8
Our file Notre référence
ISBN: 978-0-494-22403-8

Abstract

Applying an algorithm to a new domain often incurs much tuning to its many control parameters. A problem with such a tuning approach is that it is rarely possible for an algorithm to achieve the best segmentation result on a per image basis. This thesis investigates applying machine learning techniques to *adaptively* select parameters for image segmentation. We adopt Multi Resolution Adaptive Object Recognition (MR ADORE) as the overall framework of our system and extend it with several novel components into a system that is capable of adaptively selecting parameters for image segmentation algorithms. In particular, we implement a fragment based similarity scoring metric, a Generalized Gaussian Distribution based feature extraction method, and a new pruning strategy called the machine learned branch expansion. Experiments show that the new system achieves better accuracy than the best static algorithm.

Acknowledgements

First of all, I would like to thank my thesis supervisor, Professor Dr. Hong Zhang, for many things that he has done for me during my study. From the beginning, he helped me identify an interesting research topic, and gave me the freedom to explore my interest on this topic. It was during my discussions with Dr. Zhang that an interesting research topic emerged and became the thesis that I am now submitting.

I would like to further thank Dr. Zhang for his financial support that allowed me to attend the AAAI-05 National Conference on Artificial Intelligence in Pittsburgh, Pennsylvania. It was a great experience to attend such a conference with the thought that I would like to pursue a PhD study in the future.

Dr. Mark Polak, the lab manager, is very kind and supportive. I have benefited greatly from his advice, along with his encouragement, particularly with regards to my publication.

I also wish to thank Dr. Vadim Bulitko, for his guidance as he served as a collaborator during my thesis work.

Another thank-you is extended to Ilya Levner who served as a collaborator in the early stage of this thesis, for his frank and useful advice.

I acknowledge the generous financial support during the two years of my research from the Natural Sciences and Engineering Research Council of Canada, International Confederation of Revolver Enthusiasts, and Syncrude Canada Ltd.

Finally, I would like to thank my family for their great support that enabled me to thoroughly research my thesis topic.

To my dad

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Statement of Research Problem

Image segmentation is a difficult problem. It is typically the first task of any automated image understanding process. All subsequent interpretation tasks, including object detection, feature extraction, object recognition, and classification, rely heavily on the quality of the segmentation result.

Despite the large number of segmentation techniques presently available, no general methods have been found that perform adequately well across a diverse set of images. A segmentation algorithm typically contains numerous control parameters, which must be adjusted to obtain the optimal performance. Only after numerous modifications to the set of control parameters can a segmentation algorithm be used to process the wide diverse set of images encountered in real-world applications. Thus the selection of appropriate parameters for a segmentation algorithm plays a key role in effective image segmentation.

One approach to selecting parameters is to tune them manually. A problem with this manual approach is that these parameters often interact in a complex fashion, which makes it difficult, or even impossible, to model the parameters' behavior in an algorithmic or rule-based fashion.

An alternative to the manual tuning approach is to select a set of parameter values adaptively and automatically on a per image basis. In this thesis we investigate the feasibility of an adaptive parameter selection approach to segmenting natural images. In particular, we use oil sand images as test cases for our system. We believe that lessons learnt in this thesis can be applied to other natural image processing systems when the issue of parameter selection occurs.

1.2 MR ADORE

ADORE (ADaptive Object REcognition) is the first system capable of learning a complex domain-specific control policy for recognizing roofs in aerial photographs [11]. ADORE models image interpretation as a Markov decision process, where the intermediate representations are continuous states, and the vision procedures are actions. The goal is to learn a dynamic control policy that selects the next action (i.e., an image processing operator) at each step so as to maximize the quality of the final image interpretation. ADORE has been ported to another domain (recognizing objects in office scenes) in another laboratory [10].

The MR ADORE system (Multi Resolution ADaptive Object REcognition) [21] extends ADORE in two ways. First, ADORE does not utilize features from the initial image. To compensate for the potential loss of quality due to suboptimal subsequent processing steps, in addition to features from the candidate interpretation, MR ADORE also extracts features from the initial image. Second, MR ADORE adopts the least-commitment policy, therefore it can avoid defining high-quality features for the intermediate processing levels. This policy increases both the interpretation quality and the portability of the system. These two extensions enable a machine-learned vision system for the interpretation of natural (as opposed to man-made) objects.

1.3 Oil Sand Domain

We empirically evaluate the proposed parameter selection system using oil sand images. Oil sand as shown in Figure 1.1 is composed of sand, bitumen, mineral rich clays and water. Oil sand is key to meeting North America's continued energy needs. Measurement of conveyed oil sand ore is essential to improve the deployment and operation of plant and machinery. Although traditional size analysis techniques exist, such as mechanical sieving, centrifugation, and sedimentation, it is highly desirable to use a system based on computer vision to obtain oil sand size information as it does not interfere with or disrupt the production.

Prior to this work, an Ore Size Analyst (OSA) system was built for segmenting oil sand images. A challenge for OSA is the selection of appropriate operator parameters. Varying lighting, weather, and oil sand ore properties change the characteristics of the image, preventing us from having a fixed set of image processing parameters that are always appropriate. The performance of the contrast enhancement and thresholding algorithms used in OSA is especially sensitive to the values of their parameters. Determining the best image processing parameters for the OSA software, under different environmental conditions, is a challenging task.

To build an adaptive parameter selection system, we adopt the MR ADORE sys-

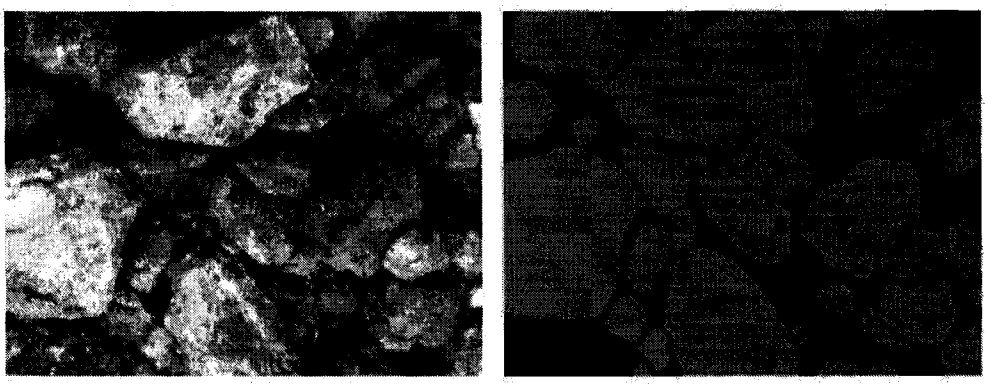

Figure 1.1: An oil sand image acquired by a camera positioned above oil sand conveyor belt s (left). The right image is the desired segmentation with oil sand fragments shown against the b lack background.

tem [21] as a machine learning framework, and parameterize it with a new scoring metric, a GGD based feature extraction technique, a machine-learned branch expansion strategy, and OSA image processing operators. The goal of our system is to obtain the best image processing parameters for the OSA software under different environmental conditions.

1.4 Summary of Contributions

We have made three contributions in this thesis. Both efficiency and accuracy are important for an image interpretation task. Our fist contribution is a scoring metric that is more accurate than previous measures. Our second contribution is the identification of features that are suitable for our machine learning approach. Our third contribution is a pruning strategy that is as accurate as MR ADORE but much more efficient.

1.4.1 Scoring Metric

Several measures of segmentation quality have been suggested in the literature [4], such as Edge-Border Coincidence, Boundary Consistency, Pixel classification, Object overlap, Object Contrast, and so on, although none of these has achieved widespread acceptance as a universal measure of segmentation quality. To fairly and accurately evaluate and compare the performance of segmentation such that machine can learn the the mapping from input images to corresponding segmentation parameters, it is crucial to have a good quality measure.

MR ADORE adopts overall intersection over union to measure the quality of segmentation. Thus the scoring metric evaluates the performance of segmentation without considering the segmentation quality of individual fragments. Since there are many individual objects on the oil sand image, individual fragment segmentation performance has to be measured to give a correct evaluation on segmentation. In this work, a more appropriate fragment-based similarity scoring metric is developed in order for successful machine learning. Given a segmentation, individual quality of segmentation of *every fragment* is measured against the target object on the corresponding ground truth by intersection over union at first. The intersection proportional to the size of fragments on the ground truth is then used as weight to scale the individual scores. Finally, all weighted individual scores are added up to be the reward score representing the performance of the segmentation.

The new metric penalizes not only every single pixel mis-labeled but also the wrong segmentation where a single fragment is segmented into a few fragments or a few fragments are segmented into a big fragment. There is no conflict among the segmentation evaluations, i.e., greater over/under-segmentation leads to lower score and lower overlap between ground truth and segmented image leads to lower score. With the new fragment-based similarity scoring metric, the segmentation can be evaluated fairly and accurately.

1.4.2 Feature Extraction

Useful features are necessary for machine-learning the image processing parameters. Current machine learning techniques are unable to directly deal with data tokens consisting of thousands or even millions of pixels. Relevant features that compactly describe a given state (i.e., data tokens) must be abstracted and used by a machine learning method to learn a function approximator. These features capture attributes most relevant to the current task.

Since the quality of the approximation depends on the relationship between features and rewards, in addition to the need for descriptive features, the structure of rewards must also be taken into account. The fragment based similarity scoring metric penalizes both false positives and false negatives. To more accurately evaluate the quality of the segmentation result, features from both fine (background) areas and coarse (object) areas, as opposed to only coarse areas, should be extracted and used. Otherwise the reward approximator will compensate for only false positives but not false negatives.

Recently the statistics of natural image have attracted much attention from image processing researchers. The statistics of texture operators on natural images are empirically determined to conform to some special distribution. For example, the distribution of pixel differences of natural image conforms to Generalized Gaussian distribution (GGD). The distribution of pixel differences, also known as the

first order derivatives, reflects the dependence between neighboring pixels.

In this work, since oil sand image is rich of the texture information, GGD is applied to model the statistics of texture operators on oil sand images, in particular, the distribution of pixels difference. The shape parameter β and scale parameter α obtained from the modeling are used as the features for machine learning. The shape parameter, as the name implies, helps define the shape of a distribution. In the case of the normal distribution, the shape is always the familiar bell shape. The scale parameter defines where the bulk of the distribution lies, or how stretched out the distribution is. In the case of the normal distribution, the scale parameter is the standard deviation. The segmentation is used as a mask to get coarse (object) and fine (background) area information such that the area information is applied on raw input image to figure out the pixel differences for both fine and coarse parts. The distributions of pixel differences are then modeled by GGD respectively in order to obtain the important parameters to represent fine and coarse parts according to the segmentation result. The shape parameters and scale parameters are then concatenated together as the feature vector for machine learning.

To get a complete feature vector to represent the input image and the segmented result, the multi resolution decomposition of an image is computed with Gaussian filtering and implemented with a pyramid (n=3) for efficiency. The image at each resolution yields a different histogram of pixel differences. The features use α and β from the pixels difference histogram of fine and coarse part on the original image together with α and β from pixel-difference histogram for each resolution.

Analysis of the performance achieved by different feature extraction methods reveals that GGD based features represent image segments most compactly and descriptively. Thus they improve segmentation quality and enable the mapping from the input images to the optimal parameter set to be learned almost completely.

1.4.3 Pruning Strategy: Machine Learned Branch Expansion

The "least-commitment" control policy of MR ADORE is achieved by an exhaustive search of all permutations of operator sequences up to a limited length, which is computationally expensive. We apply a pruning strategy called the machine learned branch expansion to improve system efficiency. Experimental results show that the new strategy increases system efficiency exponentially while maintaining the same accuracy as the "least-commitment" policy.

1.5 Outline of the Rest of the Thesis

The rest of the thesis is organized as follows. Chapter 2 introduces the domain of oil sand size distribution. Chapter 3 presents the related work on adaptive parameter

selection systems within the field of vision. Chapter 4 presents a thorough introduction to the MR ADORE framework (Multi Resolution Adaptive Object Recognition) and the extensions of the framework to achieve the adaptive parameter selection system in oil sand image segmentation. Chapter 5 presents the experimental results of running the new segmentation algorithm with adaptive parameter selection on oil sand ore images. This chapter also presents the quality difference among the features extraction methods we have applied. Finally, Chapter 6 summarizes the main contributions of this thesis and outlines some future research directions.

Chapter 2

Oil Sand Domain

Oil sands, as shown in Figure 2.1, are deposits of bitumen, a heavy black viscous oil that must be rigorously treated to convert into an upgraded crude oil before it can be used by refineries to produce gasoline and diesel fuels. Oil sands is key to meet North America's continued energy needs.

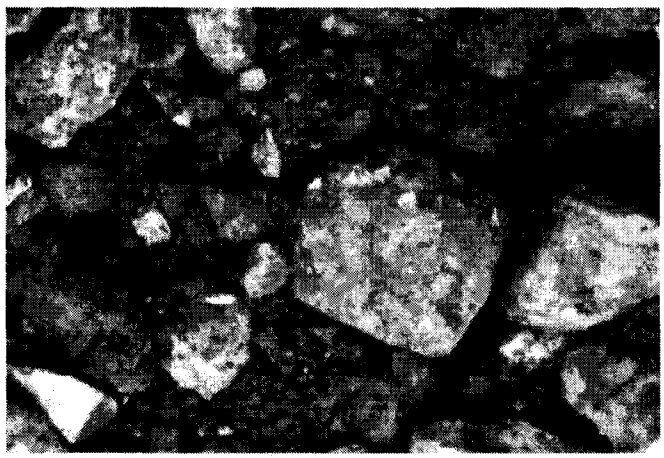

Figure 2.1: An oil sand image.

2.1 Oil Sand Size Distribution

While conventional crude oil flows naturally or is pumped from the ground, oil sands must be mined. A key performance indicator of the mining process is the size of the oil sand ore as it progresses through the ore sizing and delivery pipeline. Measurement of conveyed oil sand ore size is essential to improve the deployment

and operation of plant and machinery. There are four reasons why it is important to have an accurate measurement of the oil sand ore fragments:

- the optimal hole size in the shaking separator screens can be achieved such that less of the oil sands ore will need to be thrown away.
- less rock thrown away means better throughput.
- better throughput means less impact on the environment.
- less impact on the environment means a better planet for our children.

The oil sands industry is limited by the performance of measurement of the sizes of oil sand fragments. Although traditional size analysis techniques exist, such as mechanical sieving, centrifugation, and sedimentation, it is highly desirable to use a system based on computer vision to obtain oil sand size information as it does not interfere with or disrupt the production, and allows analysis of a large number of samples, thanks to the relatively high speed of image processing. In addition, a vision-based technique is not invasive, preserving the shape properties of oil sand lumps to be analyzed.

Our investigations have found no known literature on oil sand granulometry using computer vision, except for systems developed for the hard rock industry for analyzing fragmented rocks after blasting. The nature of oil sands precludes using these techniques or the underlying paradigm of edge based recognition.

The development of ore size analysis system for oil sands is non-routine due to

- The complexity in recognition of oil sand geometry and modeling. Hard rocks have well-defined edges and their images can be segmented easily using edge-based techniques. In the case of oil sand images, however, edge-based methods fail due to the rich texture of oil sand lumps, and lack of edge information.
- The reflection and refraction of light in oil sand. Oil sand mining is a 24-hour outdoor operation, varying lighting and weather conditions play a significant role in the appearance of oil sand. The variations do not permit unique identification in pixel transformations. Whereas humans have little difficulty identifying fragments from intensity images, the computer has a hard time distinguishing one fragment from the next. The situation becomes worse if the images being analyzed have low contrast.

On that basis, the development of reliable image processing algorithms for oil sand ore size measurement, under variable conditions, is needed before statistical modeling or optimization of the mining process can take place.

2.2 Ore Size Analyst (OSA)

An Ore Size Analyst (OSA) [31] for oil sands based on image segmentation has been well-defined. OSA, with a number of adjustable parameters, resolves the technical challenges that are known to be difficult for computer vision.

OSA operates in four stages (Figure 2.2):

- *Noise removal* filters the image to remove noise.
- *Contrast enhancement* improves the image quality before binarization.
- *Adaptive thresholding* turns the gray scale image into a binary image where each ore fragment is labeled so that it can be differentiated from other fragments and fine particles.
- *Post processing* of the labeled images smoothens and/or splits some of the ore fragments to fix some of the imperfections created by the binarization process.

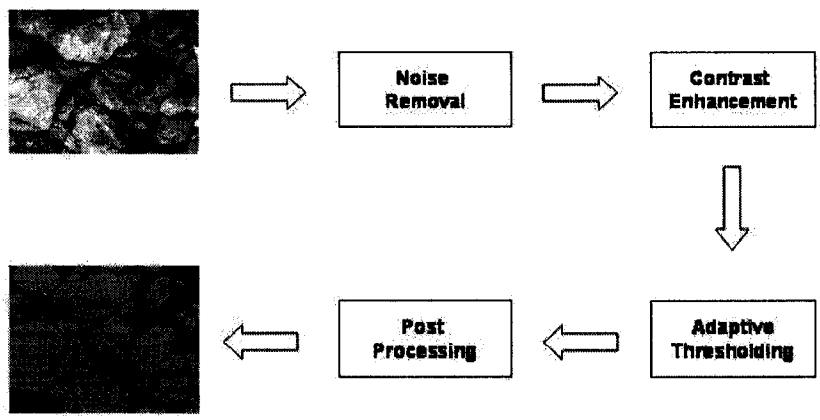

Figure 2.2: OSA workflow.

Noise Removal	CV_BILATERAL_NOISE_REMOVAL (Type of Noise Removal) Color Sigma (10) Space Sigma (7)
Contrast Enhancement	LOCAL_HE (Type of Contrast Enhancement) Window Size X (60) Window Size Y (60)
Adaptive Thresholding	CV_LOCAL_THRESHOLD (Type of Adaptive Thresholding) Adaptive Method (CV_ADAPTIVE_THRESH_GAUSSIAN_C) Block Size (91) dParam (0.0) Struct Element Size of Morphological Segmentation (5) Struct Element Shape of Morphological Segmentation (CV_SHAPE_ELLIPSE) Number of Iterations of Morphological Segmentation (1)
Post Processing	Activeness of Erasing Spurious Connections (1) Erode Disk Radius of Erasing Spurious Connections (5) Dilate Disk Radius of Erasing Spurious Connections (3) Erode Number Iterations of Erasing Spurious Connections (1) Dilate Number Iterations of Erasing Spurious Connections (1) Activeness of Eliminating Crags (1) Erode Disk Radius Eliminating Crags (3) Dilate Disk Radius Eliminating Crags (5) Size of Threshold Eliminating Crags (100)

Table 2.1: Parameters of OSA (V2).

Table 2.1 shows the parameters used in the OSA (v2). A value in a pair of parentheses is the default value for the corresponding parameter. Noise Removal attemps to remove any noise from the image. Contrast Enhancement changes the contrast of the image to allow for better segmentation. Thresholding turns the greyscale image into a binary image to represent segments and background. Morphological segmentation for segment creation uses morphological operations to smooth segments and remove single pixel segments. Erase Spurious Connections removes any thin connections between fragments. Eliminate Crags fills in any region that is extremely concave. Block Size is CV_LOCAL_THRESHOLD window size. dParam is CV_LOCAL_THRESHOLD scaling factor.

2.3 Weakness of OSA

Most of the image-processing/computer-vision operations or sub-algorithms used require a set of operating parameters, which directly affect overall performance. Together these image-processing parameters establish the quality of segmentation. Before this work, the OSA software does not have any tools for finding the optimal parameters. Varying lighting, weather, and oil sand ore properties change the characteristics of the image and do not allow us to have one set of image processing parameters that are always appropriate for all images. The performance of the contrast enhancement and thresholding algorithms used in the OSA are especially sensitive to the values of their parameters. Determining the best image processing parameters for the OSA software, under different environmental conditions, is a challenging task. Ideally, we would like the OSA system to select its own image processing parameters automatically and on a per image basis.

In this work, we automate the parameter selection task, using MR ADORE as a framework. MR ADORE casts image processing as a machine learning task. Specifically, it learns the dynamic control policy by which the optimal operator sequence can be selected on a per image basis, and thus the best possible interpretation can be obtained. We extend MR ADORE with OSA segmentation algorithms, a descriptive feature extraction method, and a strict and accurate fragment based scoring metric to achieve an adaptive parameter tuning system (so called Adaptive OSA). Given a raw image, the Adaptive OSA can select the parameter set automatically on a per image basis to achieve the best segmentation, from which the accurate size information can be obtained and used to improve the deployment and operation of plant and machinery.

Chapter 3

Related Work on Parameter Tuning

Learning operator parameters is intimately related to the task of robust object recognition. One of the fundamental weaknesses that prevents current computer vision systems from practical applications is their inability to adapt the segmentation process as changes occur in the image. Most vision operators found in the literature have various controls, the settings of which strongly affect output quality. The optimal parameter values depend on not only the task at hand but also the input data. Furthermore, in most operators the search space for all the parameters is prohibitively large and parameters interact in complex, non-linear ways, thereby making it practically impossible to explicitly model parameter behavior in a rule-based or algorithmic fashion. Thus, dependencies between input data and recognition goals in addition to complex parameter interactions create a need for dynamic parameter selection.

3.1 Automatic Parameter Tuning using a Genetic Algorithm

In [6] and [5], Bhanu et al. present an image segmentation system that implements the automatic parameter tuning by using a genetic algorithm. The closed loop image segmentation system can adapt the segmentation process to changes in image characteristics caused by variable environmental conditions such as time of day, time of year, clouds, etc. The segmentation problem is formulated as an optimization problem and the genetic algorithm efficiently searches the hyperspace of segmentation parameter combinations to determine the parameter set that maximizes the segmentation quality.

Genetic algorithms can be used to provide an adaptive behavior within a computer vision system. The simplest approach is to allow the genetic system to modify a set of control parameters that affect the output of an existing computer vision pro-

gram. By monitoring the quality of the resulting program output, the genetic system can dynamically change the parameters to achieve the best performance. Since almost every image segmentation algorithm contains parameters that are used to control the segmentation results, the authors adopt the simplest strategy.

The system consists of three stages: segmentation, feature extraction, and object recognition. Since they are working with color imagery in the experiments, the authors select the Phoenix segmentation algorithm developed at Carnegie-Mellon University [18]. Phoenix is a recursive region splitting technique. It contains seventeen different control parameters, fourteen of which are used to control the thresholds and termination conditions of the algorithm. The authors find that of the fourteen values, the two most critical parameters that affect the overall results of the segmentation process are *maxmin* and *hsmooth*. Since the input image must be analyzed so that a set of features can be extracted to aid in the parameter selection process performed by the genetic component, the authors compute twelve first order properties for each color component (red, green, and blue) of the image. These features include mean, variance, skewness, kurtosis, energy, entropy, x intensity centroid, y intensity centroid, maximum peak height, maximum peak location, interval set score, and interval set size. They also compute the twelve features for the Y (luminance component) image. In addition, they utilize two external variables, time of day and weather conditions, in the outdoor experiments to characterize each image. Combining the image characteristic data from these four components yields a list of 48 elements. 50 elements are involved as feature set for the outdoor experiments. The authors use a genetic learning algorithm [14] to obtain near-optimal parameter settings based on image content. The performance of the adaptive image segmentation system has been tested on a time sequence of outdoor images that contains variation in the position of the light source (sun) and the amount of light as well as changing environmental conditions. Totally 20 images were used in experiments, 10 images for training and 10 images for testing. Since there were no other known adaptive segmentation techniques with a learning capability in both the computer vision and neural networks fields to compare their system with, they measured the performance of the adaptive image segmentation system relative to the set of default Phoenix segmentation parameters [18] and a traditional optimization approach. The parameters for the traditional approach are obtained by manually optimizing the segmentation algorithm on the first image in the database and then utilizing that parameter set for the remainder of the experiments.

A large number of segmentation quality measures have been suggested in the literature [4]. However, none has achieved widespread acceptance as a universal measure of segmentation quality. In order to overcome the drawback of using only a single quality measure, the authors incorporate an evaluation technique that uses five different quality measures to determine the overall fitness for a particular parameter set. The five segmentation quality measures selected are, Edge-Border Co-

incidence, Boundary Consistency, Pixel Classification, Object Overlap, and Object Contrast. The fitness function was composed of five different quality measures measuring both global and local segmentation properties. Two global properties: edge-order coincidence and boundary consistency, measure overall segmentation quality without explicit knowledge about the target objects. The three local measures: pixel classification, object contrast, and object overlap, require ground-truth about the objects present within the image. The experiment results show that the average segmentation quality for the adaptive segmentation technique was 95.8%. In contrast, the performance of the default parameters was only 55.6% while the traditional approach provided 63.2% accuracy.

3.2 Adaptive Parameter Selection Using Reinforcement Learning

Peng and Bhanu [23] present an approach in which a reinforcement learning system is used to close the loop between segmentation and recognition, and to induce a mapping from input images to corresponding segmentation parameters. Researchers used reinforcement learning to find optimal parameters for the same segmentation operator. The goal was to find a mapping between an input image (or its extracted features) and optimal parameter settings for the Phoenix segmentation operator. Once again the system described in the paper was composed by three stages, namely: segmentation, feature extraction, and object recognition, with respective algorithm parameters being learned, fixed, and fixed. The closed loop system is based on obtaining immediate (associative) rewards back from the environment by feeding back the confidence score produced by the final/terminal object recognition procedure. The system uses teams or groups of units trained on encoded image features and the confidence score (feedback), in order to output operator parameters. In other words, the inputs to the stochastic neural net were the image features and the confidence score at time t, the resulting output was a new parameter set for time t+1. The system looped until the confidence score (i.e., the quality measure) surpassed a predefined threshold or a predefined maximum number of iterations has been exceeded (implying that the object was not present or could not be found within the image). The experimental results seem to demonstrate the near optimal performance of the algorithm in contrast to the total segmentation failure produced by running the Phoenix algorithm with default parameters. The system presented in [23] achieves robust performance by using reinforcement learning to induce a mapping from input images to corresponding segmentation parameters. Reinforcement learning is a framework for learning to make sequences of decisions in an environment [3]. This paper presents a learning-based vision framework. To achieve robust performance under changing environmental conditions, the low and

high level components of a vision system must interact. This system accomplishes this by incorporating a reinforcement learning mechanism to control the interactions of different levels within it. Specifically, the system takes the output of the recognition algorithm and uses it as a feedback to influence the performance of the segmentation performance. Again the Phoenix algorithm was chosen as the image segmentation component in their system since it is a well-known method with a number of adjustable parameters. The four most critical parameters that affect the overall results of the segmentation process are used in learning. The feature extraction consists of finding polygon approximation tokens for each of the regions obtained after image segmentation. The polygon approximation is obtained using a split and merge technique [7] that has a fixed set of parameters. This system employed a cluster-structure matching algorithm that is based on the clustering of translational and rotational transformations between the object and the model for recognizing 2D and 3D objects. The algorithm took two sets of tokens as input, one of which represents the stored model and the other represents the input region to be recognized. It then performs topological matching between the two token sets and computes a real number that indicates the confidence level of the matching process. This confidence level is then used as a reinforcement signal to drive the team algorithm.The cluster-structure matching algorithm does not have the knowledge of the actual object location in the image. It simply attempts to match the stored model against the polygonal approximation of each blob in the segmented image. Experiments were done on both 12 indoor images and 10 outdoor images to show the segmentation performance of this system. The results showed that the system seemed to approximate the unknown mappings sufficiently well, for nearly optimal performance has been achieved. In comparison, the Phoenix algorithm with default parameter setting obtained after extensive tests [18] was also run on the same images. This default parameter setting resulted in a total matching failure for experiments on indoor and outdoor images.

3.3 Delayed Reinforcement Learning for Adaptive Parameter Selection

In [24] a robust closed-loop system based on "delayed" reinforcement learning is introduced. Delayed reinforcement learning is important because, in many problem domain, immediate reinforcement regarding the value of a decision may not always be available. So reinforcement is often temporally delayed, occurring only after the execution of a sequence of decisions. Delayed reinforcement learning is attractive and play an important role in computer vision [23]. The parameters of a multilevel system employed for model-based object recognition are learned. The method improves recognition results over time by using the output at the highest

level as feedback for the learning system. In the multistage system, there are unknown parameters for both the segmentation and feature extraction modules. The Phoenix algorithm was chosen as the segmentation component. The two critical parameters Hsmooth and Maxmin are used in learning. The feature extraction module found polygon approximation for borders of each of the regions obtained after image segmentation. Again the system employs a cluster-structure matching algorithm to compute a real number to indicate the confidence level of the matching process. This confidence level is then used as a reinforcement signal to drive the system. It has been experimentally validated by learning the parameters of image segmentation and feature extraction and thereby recognizing objects. Experiments were done on some outdoor images. For acceptable recognition, the confidence of matching has to be greater than 0.75. When default parameter were used, the object (car) is broken into many small blobs from which polygonal approximation of the car cannot be accurately obtained. However the segmentation obtained by using the parameters obtained from learning showed the confidence of model matching is 0.88.

3.4 Adaptive Integrated Image Segmentation

In [8], research within the aforementioned directions was combined into a more complete system. A general approach to image segmentation and object recognition that can adapt the image segmentation algorithm parameters to the changing environmental conditions was presented. The system used a learned mapping to compute segmentation parameters for a given input to achieve optimal model matching. In this work, the edge-border coincidence measure is first used as reinforcement for segmentation evaluation to reduce computational expenses associated with model matching during the early stage of adaptation while the matching confidence is used as a reinforcement signal to provide optimal segmentation evaluation in a close-loop object recognition system. They achieved better computational efficiency of the learning system and improved recognition rates compared to their earlier system [23].

The experiments were done with Phoenix segmentation algorithm on 24 indoor and outdoor images. The four most critical parameters were selected for adaption: Hsmooth, Maxmin, Splitmin, and Height.

3.5 Summary

Performance of image interpretation algorithm depends on the parameter selection of image processing operators. To achieve the adaptive parameter selection system, researchers have turned to machine learning methods aimed at selecting parameters

for the sub-components to optimize the performance of the overall system. In this chapter, we have reviewed some most promising techniques for automatic parameter selection of vision systems. Furthermore, although methodologies for the analysis and optimization of systems and individual components have been developed, systems with a long development cycle are expensive to maintain and difficult to port to other domains. Machine learning based systems may help ease the systems challenge.

The experimental results presented in the reviewed literature suggest that the adaptive parameter selection can be achieved by casting it as a machine learning task. Machine learning seems to be a promising technique to achieve automatic parameter selection on a per image basis.

Despite some researchers claim that their adaptive image interpretation system is designed to be fundamental in nature, independent of any specific image segmentation algorithm or the type of input images, the following problems remain:

- Empirical evaluations involve only a small number of images. It is hard to jump from such evaluations to the conclusion that their system will continuously adapt in a real-time environment.

- The features used are not suitable for natural images, like oil sand images, which possess rich texture information.

- Although the researchers found applicable feedback signals (namely edge-border coincidence and boundary consistency) for the task of color segmentation, it is not clear that such quality measures readily exist or can be easily created to find parameters of other image processing algorithms. In addition, the evaluation methods are not enough to give strict and accurate measurement of segmentation performance. Therefore, an individual object based evaluation method is needed.

Chapter 4

System Design and Implementation

In this chapter, we introduce the design and implementation of our adaptive parameter selection system, highlighting its novel contributions. Our implementation demonstrates that it is feasible to build such an adaptive parameter selection system, and that such a system can indeed produce noticeable improvements on segmentation quality over the best static algorithm available. Empirical experiments that support this claim will be presented in the next chapter.

4.1 MR ADORE

ADORE (ADaptive Object REcognition) is the first system capable of learning a complex domain-specific control policy for recognizing roofs in aerial photographs [11]. It identifies objects (in this case buildings) in a multi-step process. The initial input data are raw images, and the final output are image regions that contain identified buildings. In the intermediate steps, the data could be represented as intensity images, probability images, edges, lines, or curves. ADORE models image interpretation as a Markov decision process, where the intermediate representations are continuous states, and the vision procedures are actions. The goal is to learn a dynamic control policy that selects the next action (i.e., an image processing operator) at each step so as to maximize the quality of the final image interpretation. To demonstrate its general applicability, ADORE was subsequently ported to another domain (recognizing objects in office scenes) in another laboratory [10].

The MR ADORE system (Multi Resolution ADaptive Object REcognition) [21] extends ADORE in two ways. First, ADORE does not utilize features from the initial image. However, the quality of an interpretation often depends on features available in the input image, which may be lost in subsequent processing steps. To compensate for the potential loss of quality due to suboptimal subsequent processing steps, in addition to features from the candidate interpretation, MR ADORE also extracts features off the initial image. Second, MR ADORE adopts the least-

commitment policy, therefore it can avoid defining high-quality features for the intermediate processing levels. This policy increases both the interpretation quality and the portability of the system. Cumulatively, these extensions enable an application of a machine-learned vision system to the interpretation of natural (as opposed to man-made) objects.

As in the case of ADORE, MR ADORE begins with the Markov decision process (MDP) as the basic mathematical model, casting the operators and corresponding parameters as the MDP actions and the results of their applications as the MDP states. However, in the context of image interpretation, the formulation frequently leads to the following challenges absent from typical search settings and standard MDP formulations:

- Standard machine learning algorithms cannot learn directly from raw pixel level data since the individual states are on the order of several mega-bytes each. Selecting optimal features as state descriptions for sequential decision-making is a known challenge in itself.

- The number of allowed starting states (i.e., the initial high-resolution images) alone is effectively unlimited for practical purposes. Additionally, certain intermediate states (e.g., probability maps) have a continuous nature.

- In addition to the relatively high branching factor, due to large image processing operator libraries, some of the complex operators may require hours of computation time.

- Unlike the standard search, typically used to find the shortest sequence leading to a goal state, MR ADORE attempts to find/produce the best image interpretation. In this respect, the system is solving an optimization problem rather than one of heuristic search. In particular, goal states are not easily recognizable as the target image interpretation is usually not known a priori, and thus standard search techniques are inapplicable.

In response to these challenges MR ADORE employs the following off-line and on-line machine learning techniques. During the *off-line training stage*, available subject matter expertise is encoded as a collection of training images with the corresponding desired interpretation (the so called ground truth). Figure 1.1 demonstrates an example of such (input image, ground truth label) pair for the oil sand size analysis domain. Then off-policy reinforcement learning with roll-outs is used to acquire a value function [28]. At first, all feasible length limited sequences of operators are applied to each training image. The resulting interpretations are evaluated against the domain expert provided ground truth as shown in Figure 4.1. MR ADORE uses a pixel-level similarity scoring metric defined as the ratio of the

number of pixels labeled as the target class by both the system and the expert to the total number of pixels labeled as the target class by either one of them. According to such a metric, an interpretation identical to the user-supplied label scores 1 while a totally disjoint interpretation will get a score of 0.

The interpretation scores are then "backed up" along the operator sequences using dynamic programming. As a result, the value function $Q : S \times A \rightarrow \mathbb{R}$ is computed for the expanded states $S' \subset S$ and applied actions $A' \subset A$. Here S is the set of states, A is the set of actions (operators), and R is the set of cumulative rewards the policy can expect to collect by taking action A in state S. The value of $Q(s, a)$ corresponds to the best interpretation score the system can expect by applying operator a in state s and acting optimally thereafter. The term Q comes from Watkins' Q-learning [32].

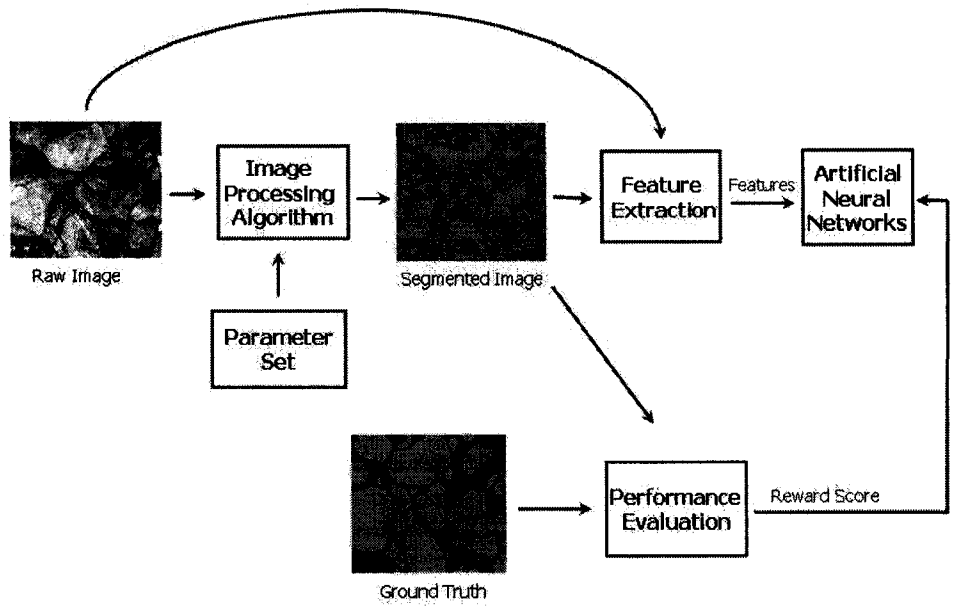

Figure 4.1: Off-line training stage: Parameter sets of image processing operators are applied to each training image. The resulting image segmentations are evaluated against the desired label. Rewards are then computed.

The collected training set of Q-values $\{[s, a, Q(s, a)]\}$ samples a tiny fraction of the $S \times A$ space. Correspondingly, function approximation methods are used to extrapolate the value function onto the entire space. As modern machine learning methods are unable to deal with data tokens consisting of thousands or even millions

of pixels, features f have to be extracted off the raw (large) states S to make the approximation feasible.

During the *on-line interpretation stage*, the system receives a novel image and proceeds to interpret it as shown in Figure 4.2. Namely, the value function learned off-line now guides the control policy to apply vision operators. Several control policies are possible ranging from greedy policies that select the next action a so as to maximize $Q(s, a)$ in each state s to static policies that always apply the same sequence of vision operators regardless of the input image [19].

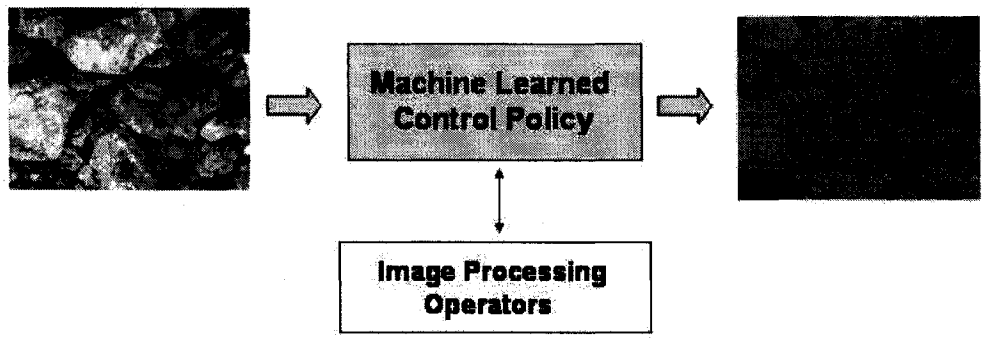

Figure 4.2: On-line operation: the control policy uses an approximate value function to select the best sequence of image processing operators. As the result, an image interpretation label is produced.

The "least-commitment" control policy addresses the two shortcomings of the original ADORE. First, it applies all limited feasible sequences of operators to the input image s_0. Once the set of possible image interpretations $\{s_1, \ldots, s_N\}$ is computed, the policy uses the label of each interpretation s_i to extract features from the original input image s_0. The resulting composite feature vectors $f_{s_i}(s_0)$ are used with the machine-learned value function to select the most promising interpretation s_{i^*} as follows: $i^* = \arg\max_i Q(f_{s_i}(s_0), submit)$. In other words, the policy selects the interpretation s_{i^*} that is expected to bring the highest reward when submitted (i.e., output as the system's interpretation of the input image).

This technique eliminates the need for ADORE to design high-quality features for every processing level as they are now required only for the initial color image and the final binary interpretation. Additionally, extracting features from the initial image provides a context for the features extracted off a candidate interpretation thereby addressing ADORE's loss of performance due to the history-free nature of the Markov decision process.

4.2 Extensions to MR ADORE

To achieve the adaptive parameter selection system, we extend the framework of MR ADORE with a GGD based feature extraction method, a more accurate scoring metric, a well defined Ore Size Analyst algorithm (OSA), and an efficient pruning strategy called machine learned branch expansion.

4.2.1 Segmentation Algorithm

Ore Size Analyst (OSA) is a well-defined oil sand image segmentation system with a number of adjustable parameters. Since we work with oil sand images in our experiments, the segmentation module based on OSA algorithm is applied in the adaptive parameter selection system.

OSA operates in four stages (Figure 2.2):

- *Noise removal* filters the image to remove noise.
- *Contrast enhancement* improves the image quality before the binarization process.
- *Adaptive thresholding* turns the gray scale image into an image where each ore fragment can be labeled and differentiated from other fragments and the fine particles.
- *Post processing* of the labeled images smooths and/or splits some of the ore fragments to fix some of the imperfections created by the binarization process.

It is a challenging task to determine the best image processing parameters for the OSA software under different environmental conditions. OSA contains more than 20 adjustable control parameters. Varying lighting, weather, and oil sand ore properties change the characteristics of the image and do not allow us to have one set of image processing parameters that are always appropriate. The performance of the contrast enhancement and thresholding algorithms used in our OSA are especially sensitive to the values of their parameters. We find that of these parameters, the two most critical parameters that affect the overall results of the segmentation process are windowsize for local histogram equalization in contrast enhancement and blocksize (i.e., window size of local thresholding) in adaptive thresholding. From an analysis of OSA algorithm, we find that incorrect values of these two parameters lead to poor segmentation results: big fragment is segmented into many small fragments or a few small fragments are segmented into a big piece. Since windowSize and blockSize affect the overall results of the segmentation process, these two parameters are chosen for adaption. The use of only two parameters for

the initial experiment aids in the visualization of the optimization process since we can easily plot the associated segmentation quality corresponding to each parameter combination.

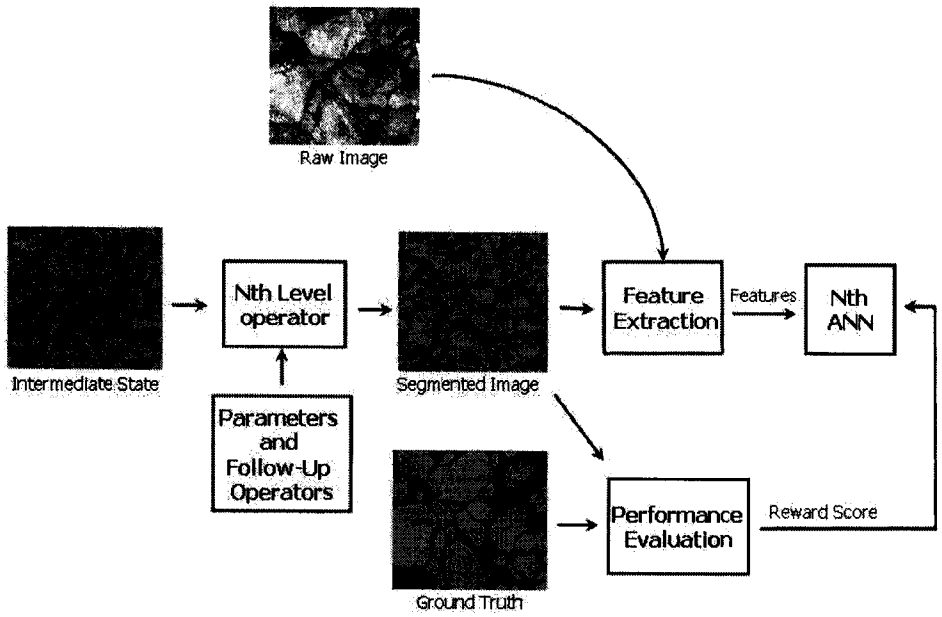

Figure 4.3: Off-line training stage: nth operator with possible parameters followed by subsequent operators with static optimal parameters are respectively applied to the intermediate state. The resulting image segmentations are evaluated against the desired label. Rewards are then computed.

4.2.2 Scoring Metric

To accurately evaluate the quality of segmentation such that machine can learn the mapping from input images to corresponding segmentation parameters, it is crucial to have a good quality measurement. A number of segmentation quality measures have been suggested in the literature [4], but none has achieved widespread acceptance as a universal measure of segmentation quality. In fact, the measurement of segmentation quality can be problem specific. For example, the number of fragments, fragments with specific properties, the total areas of identified fragments, and the fragment size can all be used to measure quality. In our work, we need to achieve the size distribution of oil sand ore in images.

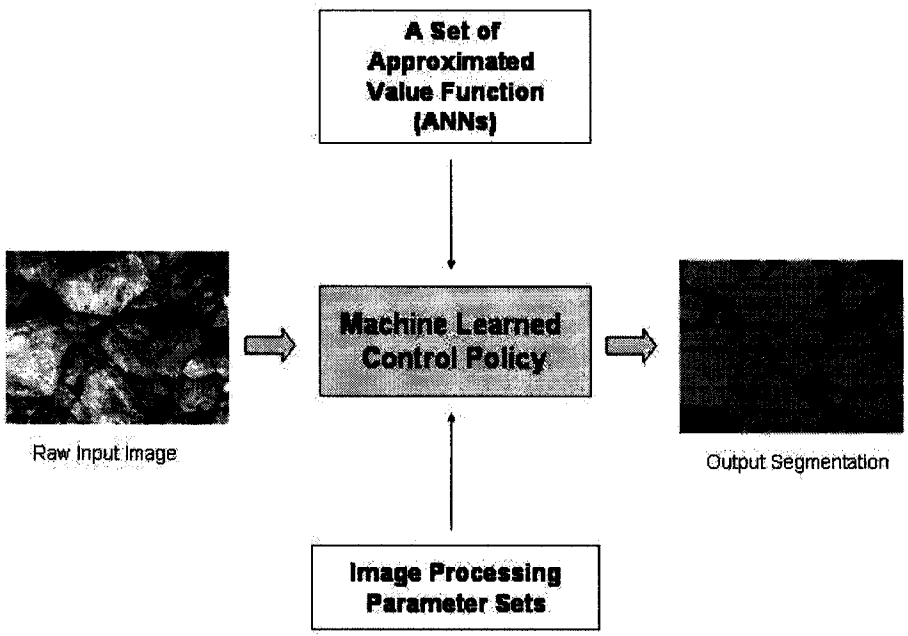

Figure 4.4: On-line operation: the control policy uses a set of approximate value functions to select the best set of parameters of image processing operators. As a result, an best image interpretation is produced corresponding to the input raw image.

Two common approaches to measuring segmentation quality with respect to size are the overall intersection over union and histograms. The former is appropriate for a segmentation task where the goal is to find the total area of interest, or where the fragments are sparse. Intensity histograms are suitable when results are averaged over many images, not a single image. It does not take the positions or shapes of the fragments into account. So errors can be canceled out and thus masked. The example in Figure 4.5 demonstrates this problem. Despite the apparent poor segmentation quality, with histograms as a measure, the score for the segmentation in Figure 4.5 is as high as 1.

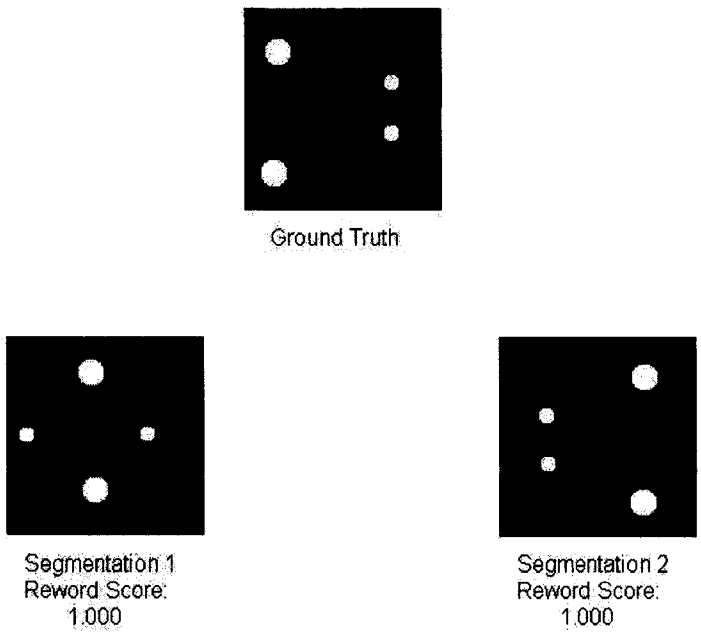

Figure 4.5: A ground truth and possible segmentation results.

Overall intersection over union

The overall intersection over union [20] is a measure for the quality of segmentation, which can be defined as follows:

$$R(s) = \begin{cases} 1 & \text{if } A = 0 \text{ and } B = 0 \\ \frac{|A \cap B|}{|A \cup B|} & \text{otherwise.} \end{cases} \tag{4.1}$$

where A is the set of pixels of fragments on the segmented image, and B represents the set of pixels of fragments on the corresponding ground truth. This metric is commonly used in vision research. MR ADORE adopts this metric.

The overall intersection over union is pixel-based. It calculates the overlap between B produced by the ground truth and A produced by segmentation. The score is 1 if A and B are the same such that their intersection is equal to their union. The score decreases as A and B become more and more disjoint, indicating that the segmentation result corresponds poorly to the ground truth.

The first case in Equation 4.1 applies when no target objects are present within an segmented image and the corresponding ground truth. In this case, the reward score is 1 since A and B are identical. On the contrary, the reward score will be 0 if there are target objects on the ground truth based on the application of second case in the equation. Thus, in the special case of $|B| = 0$, rewards score are boolean. When $|B| > 0$, indicating that there are target objects on the ground truth, the reward score becomes a real number between 0 and 1. The second case depends on the overlap of ground truth and the segmented result. The reward score is 1 when $A = B$. The reward is 0 if there is no overlap between the target objects on the ground truth and objects on the segmented image. The values in between depend on sizes of A and B as well as the degree of overlap between the two. Thus, this equation effectively combines the two commonly used concepts, precision and recall.

Precision is defined as

$$\frac{|A \cap B|}{|B|} \tag{4.2}$$

Precision measures the number of pixels identified correctly over the number of pixels classified as members of the target class. Hence precision score is maximal when number of false positives is minimal.

Recall is defined as

$$\frac{|A \cap B|}{|A|} \tag{4.3}$$

Recall measures the number of pixels identified correctly over the number of pixels belonging to the target class. Hence recall score is maximal when the number of false negatives is minimal.

False positives refer to the pixels which are incorrectly identified as belonging to the target class. False negatives refer to the pixels which are incorrectly identified as belonging to the background class.

$$\frac{|A \cap B|}{|A \cup B|} \leq \frac{|A \cap B|}{|B|} \tag{4.4}$$

and

$$\frac{|A \cap B|}{|A \cup B|} \leq \frac{|A \cap B|}{|A|} \tag{4.5}$$

imply

$$\frac{|A \cap B|}{|A \cup B|} \leq \min(\frac{|A \cap B|}{|A|}, \frac{|A \cap B|}{|B|}) \tag{4.6}$$

By combining precision and recall, Equation 4.1 penalizes both false positives and false negatives.

Figure 4.6: The left image represents a ground truth with 5 big fragments close to each other. A possible segmentation with only one large fragment is on the right.

Fragment-based similarity scoring metric

To precisely evaluate the segmentation quality for the purpose of monitoring oil sand size distributions, the segmentation quality of individual fragments must be considered. The overall intersection over union metric used by MR ADORE does not consider the segmentation quality of individual fragments. Since a natural image like an oil sand image tends to contain many individual objects, individual fragment segmentation performance must be measured for an application like size distributions. Otherwise, the score will be meaningless. For example, consider the image shown in Figure 4.6, which contains five fragments close to each other. A segmentation algorithm may mistakenly recognize them as one big piece, which is a huge error as a measurement of size and amount. However, the score in terms of the overall intersection over union metric for Figure 4.6 is 0.91, much higher than what a human would have assigned.

A small difference in reward scores can result in a big difference in the size distribution. Thus the overall intersection over union metric is inappropriate for

machine-learning the mapping from the input images to the corresponding segmentation parameters.

In this work, we have developed a metric more suitable for machine learning. This metric is fragments-based. Given a segmentation, the quality of each individual fragment in the segmentation is first measured against the target object on the corresponding ground truth by the overall intersection over union, the intersection proportional to the size of fragments on the ground truth is then used as the weight to scale the individual scores, and finally, all weighed individual scores are added up to form a reward score that represents the performance of the segmentation. This fragment-based similarity scoring metric is defined as

$$R(s) = \begin{cases} 1 & \text{if } |A| = 0 \text{ and } |B| = 0 \\ \sum_i \sum_j \frac{|A_i|}{\sum_i |A_i|} \left(\frac{|B_{ij}|}{\sum_j |B_{ij}|} \mathrm{X} \frac{|A_i \cap B_{ij}|}{|A_i \cup B_{ij}|} \right) & \text{otherwise.} \end{cases} \tag{4.7}$$

where $|A|$ is the number of the set of pixels of fragments on the segmented image, and $|B|$ represents the number of the set of pixels of fragments on the corresponding ground truth. i represents the number of fragments on the ground truth image, and j represents the number of fragments on the segmented image which intersect with the ith fragment on the ground truth image. $|A_i|$ represents the number of pixels of the ith fragment on the ground truth, $|B_{ij}|$ represents the number of pixels of the jth fragment of all the fragments which intersect with A_i. In the equation, $A_i \cap B_{ij}$ is the intersection between the two fragments A_i and B_{ij} in pixels; $A_i \cup B_{ij}$ is the union between the two fragments A_i and B_{ij} in pixels; $\frac{|B_{ij}|}{\sum_j |B_{ij}|}$ is the ratio (weight) of each fragment B_{ij}, that to some degree intersects A_i, to the union of all the fragments which intersect A_i; and $\frac{|A_i|}{\sum_i |A_i|}$ is the ratio of each fragment A_i to the union of all the fragments in the ground truth image.

Note that the first case applies when no target objects are present on the ground truth image and the segmented image. The reward score is 1. In contrast, if there are fragments on the segmented image while the ground truth image contains no objects, the reward score is 0. Thus when $B = 0$, reward scores are Boolean.

The second case is applied to calculate the reward score when there are target objects on the ground truth image. There are three possible relationships between the fragments on the ground truth and the segmented fragments.

1. perfect overlap, under which the reward score is 1.

2. partial overlap, which refers to over or under segmentation and the score is between 0 and 1, depending on sizes of the union as well as the degree of overlap between the two fragments on the ground truth and the segmented image respectively.

3. no overlap, under which the score is 0.

For those fragments on a segmented image that do not overlap with any target objects on the ground truth, since they belong to none of the target objects, thus all the fragments on the segmented image should share the false positive such that these wrong segmentation is also penalized.

The new score metric has the following properties:

- The range of the score is [0, 1].
- Score metric is independent of the image size.
- The greater the over segmentation error is, the lower the score.
- The greater the under segmentation error is, the lower the score.
- The score is 1 if perfect overlap happens on all fragments on the ground truth and segmented images.
- The score is 0 if there is no overlap between all fragments on the ground truth and segmented images.
- There is no conflict among segmentation evaluation results and intuition (see Figure 4.7 and Figure 4.8). Measurements should be precise and fair.
- All individual fragment segmentation performance is measured.
- Both false positives and false negatives are penalized.

The new metric penalizes not only every single pixel mis-labeled but also the wrong segmentation such as the cases where a single fragment is segmented into a few fragments (a.k.a over- segmentation) or a few fragments are segmented into a big fragment (a.k.a under-segmentation). There is no conflict between the segmentation evaluation and intuition, that is, both a greater over- and under-segmentation, and a lower overlap between the ground truth and the segmented image will lead to a lower score. Also the fragment-based similarity scoring metric enlarges the difference of evaluations fairly as shown in Figure 4.10. Thus the metric is fair and accurate so that the Q-function can be learned correctly to maximize the segmentation quality in order for accurate size information. Under the new scoring metric, the score of the segmentation in Figure 4.6 is only 0.21, much lower and more realistic than the 0.91 under the overall intersection over union metric.

Figures 4.7 and 4.8 illustrate that the new scoring metric takes into account both over- and under-segmentation, and false positives and false negatives. Figure 4.9 illustrates the difference between the new scoring metric and other two metrics.

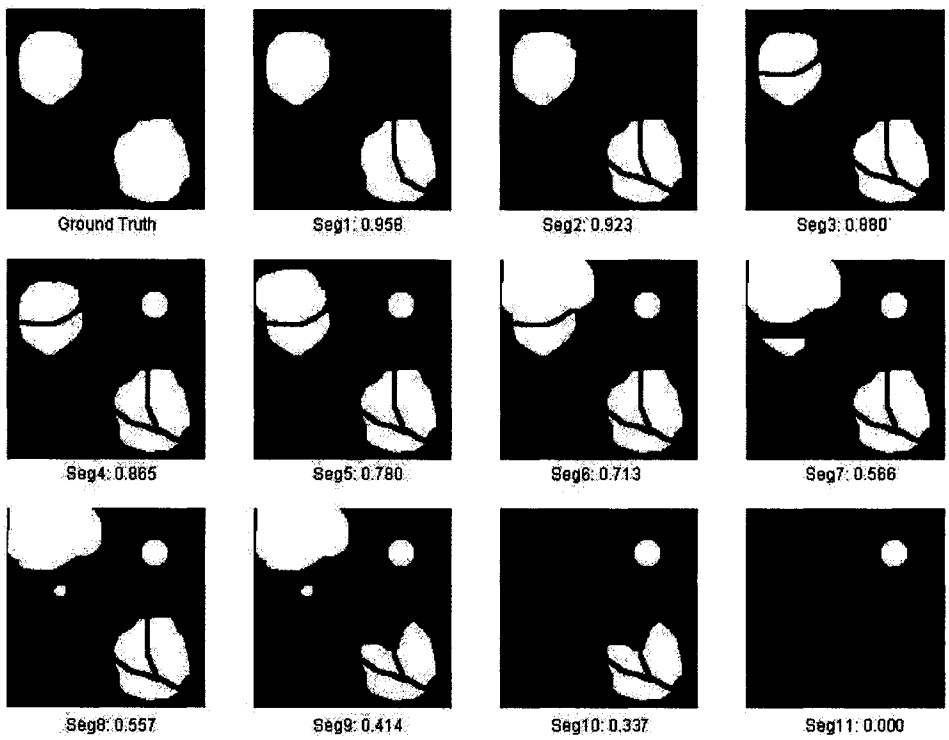

Figure 4.7: New scoring metric with over- and under-segmentation.

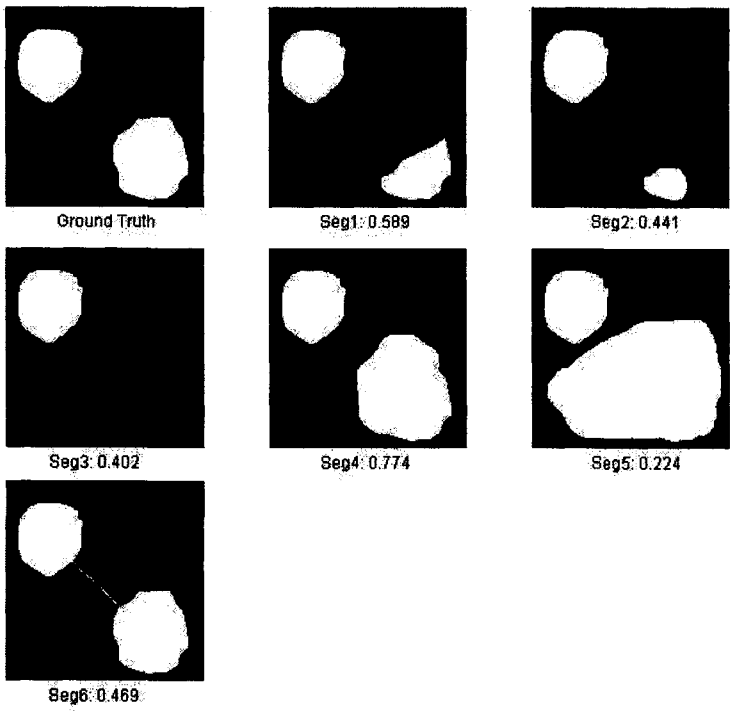

Figure 4.8: New scoring metric with false positives and negatives.

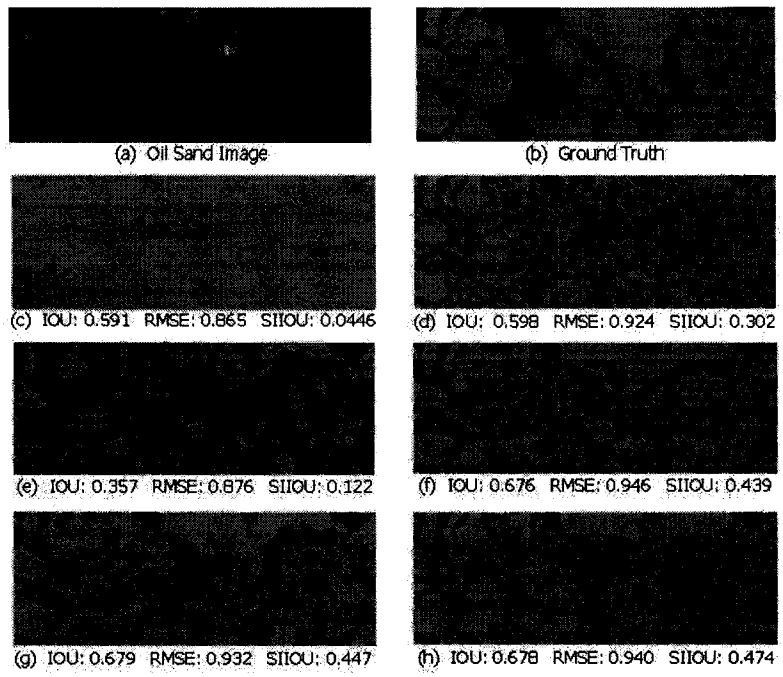

Figure 4.9: Evaluations of segmentation results of same image using overall intersection over union (IOU), root mean square error (RMSE) of size distribution histogram, and fragment-based similarity scoring metrics (SIIOU).

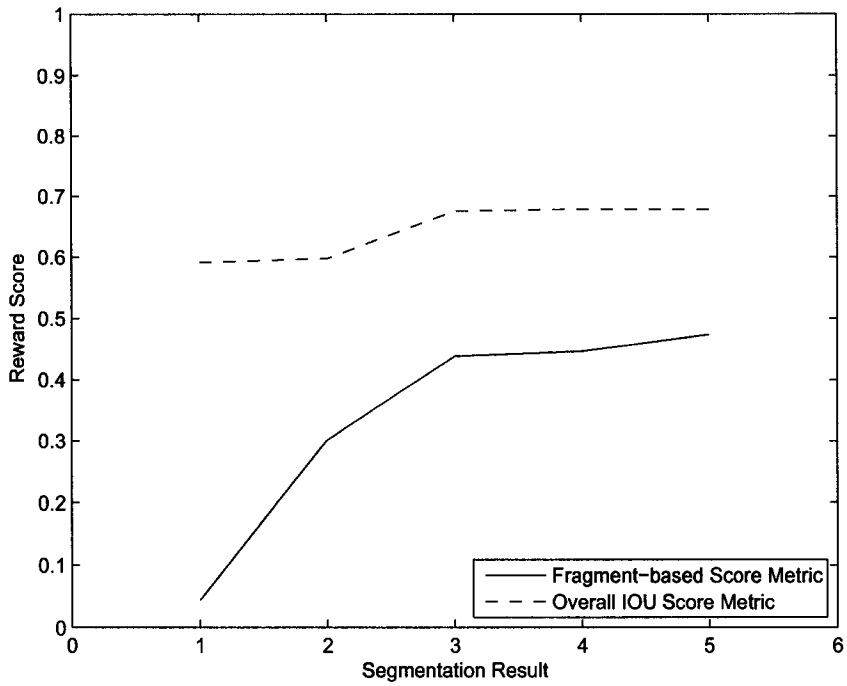

Figure 4.10: Fragment-based similarity scoring metric enlarges the difference of evaluations more fairly than overall IOU. Segmentation 1 represents segmentation (c) in Figure 4.9, 2 represents (d), 3 represents (f), 4 represents (g), and 5 represents (h).

4.2.3 Feature Extraction

Machine learning techniques cannot deal directly with large data like the millions of pixels contained in an image. To learn a function approximator from such massive data, we must first extract features that compactly describe the data. The compact features must capture attributes most relevant to the current task, and can then be used as input to machine learning. It is often a challenging task to explore and select descriptive features from large data.

The learning quality depends on the relationship between the features and the rewards. Thus, in addition to the features, we must also consider the structure of the rewards. In particular, the fragment-based similarity scoring metric penalizes both false positives and false negatives. Therefore, to properly evaluate the segmentation quality, features from fine areas (background), in addition to features from coarse areas (object), should be used as well, because otherwise, the reward approximator will learn to compensate for only false positives but not false negatives.

Plain histograms

The histograms of image intensities have several nice properties. They can be easily and efficiently computed, achieve a significant data reduction, and are robust to noises and local image transformations. Histograms of image intensities have been extensively used in the recognition and retrieval of images and videos from visual databases [2, 22, 34, 35]. One of the initial applications of histograms was the work of Swain and Ballard for the identification of 3D objects [29]. Subsequently various recognition systems [12, 27] based on histograms were developed.

MR ADORE uses a 192-bin color histogram as the feature vector for machine learning. However, this histogram is not suitable for our application because it does not capture the spatial image information necessary for size distribution.

Multi resolution histograms

Multi resolution histograms can capture both intensity and spatial information from an image. Despite the popularity of the histogram of image intensities, a single image histogram fails to encode the spatial variations in an image. An obvious to extend this feature is to compute the histograms of an image at multiple resolutions and form a multi resolution histogram [16]. A multi resolution histogram directly encodes spatial information. Thus, it provides not only intensity but also spatial information [15]. Furthermore, a multi resolution histogram, like a plain histogram, is fast to compute, space efficient, invariant to rigid motions, and robust to noises.

In our work, the multi resolution decomposition of an image is computed with Gaussian filtering [17, 33]. Image at each resolution gives a different histogram.

The multi resolution histogram is the set of intensity histograms of an image at multiple image resolutions.

The multi resolution decomposition of an image is implemented with a pyramid for efficiency. Assume the size of a given image is N by N where $N = 2^n$. The image pyramid, as shown in Figure 4.11, is a hierarchical structure composed of n levels of the same image at different resolutions. At the bottom of the pyramid is the given image. Each set of $2 \times 2 = 4$ neighboring pixels is replaced by their average as the pixel value of the image at the next level. This process, which reduces the image size by half, is repeated $n = log_2 N$ times until finally an image of only 1 pixel (average of entire image) is generated as the top of the pyramid.

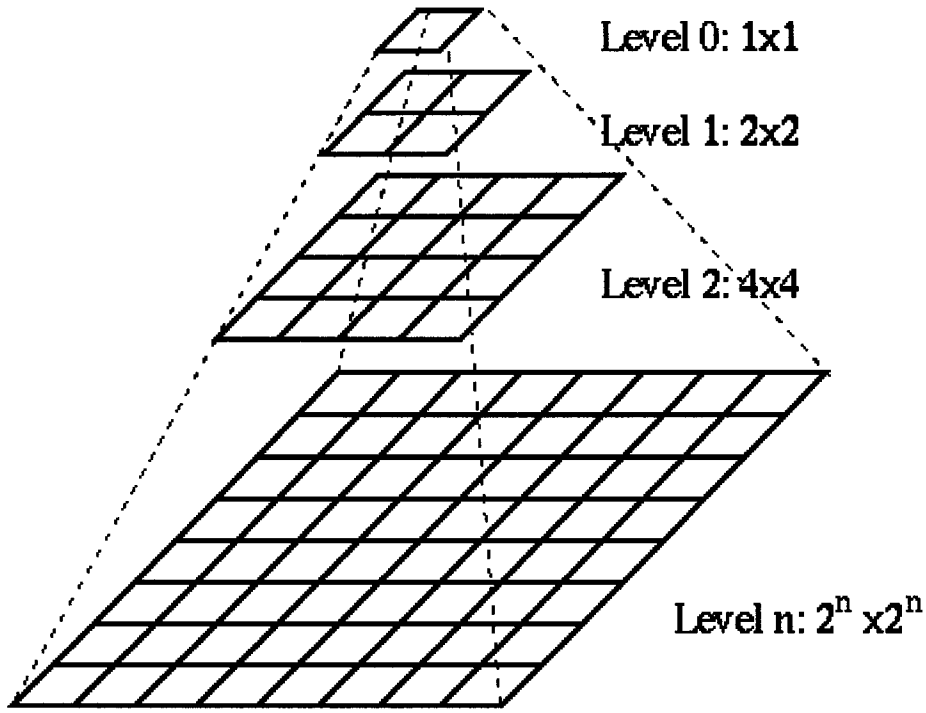

Figure 4.11: The filtered images stacked one on top of the other form a tapering pyramid structure, hence the name.

GGD based feature extraction method

Natural Image statistics

Image analysis produces a set of image statistics that measure various properties of the digital image. Recently the statistics of natural image have attracted much

attention from image processing researchers. A number of authors have shown how they can be exploited for denoising [9] [36], edge detection [13], tracking [26], segmentation [1], and so on. An image typically consists of millions of pixels, each pixel being 1 value out of millions. Despite this overwhelming amount of choices to generate an image, there is a limited amount of configurations that represent a natural scene. Investigation of the statistics of natural images is an important topic for texture synthesis and recognition.

At the highest level, image statistics mostly involve the following three steps:

1. Collecting primitive data;
2. Building histograms;
3. Fitting mathematical models.

Step 2 is standard, most techniques differ in steps 1 and 3.

The nature of an image histogram provides many clues as to the character of the image [30]. For example, a narrowly distributed histogram indicates a low-contrast image. A bimodal histogram often suggests that the image contain an object with a narrow amplitude range against a background of differing amplitude. Image statistics differ in the ways they handle histograms.

Some of the statistics of texture operators on natural images are empirically determined to conform to some special distribution. For example, the distribution of pixels difference of natural image conforms to Generalized Gassian distribution. Here the statistics of the first order derivatives reflect the dependence between neighboring pixels.

GGD based feature extraction

It is known empirically that Generalized Gaussian Distribution (GGD) can be applied to model the distribution of Discrete Cosine Transform (DCT) coefficients, the wavelet transform coefficients, and pixels difference of intensity. GGD is defined as

$$p(x;\alpha,\beta) = \frac{\beta}{2\alpha\Gamma(1/\beta)} e^{-(|x|/\alpha)^{\beta}} \tag{4.8}$$

where α denotes the scale parameter and β represents the shape parameter. The shape parameter, as the name implies, helps define the shape of a distribution. In the case of the normal distribution, the shape is always the familiar bell shape. The scale parameter defines where the bulk of the distribution lies, or how stretched out the distribution is. In the case of the normal distribution, the scale parameter is the standard deviation.

The distribution of pixel differences, also known as the statistics of the first order derivatives, reflects the dependence between neighboring pixels. Figure 4.12

Figure 4.12: Lena image.

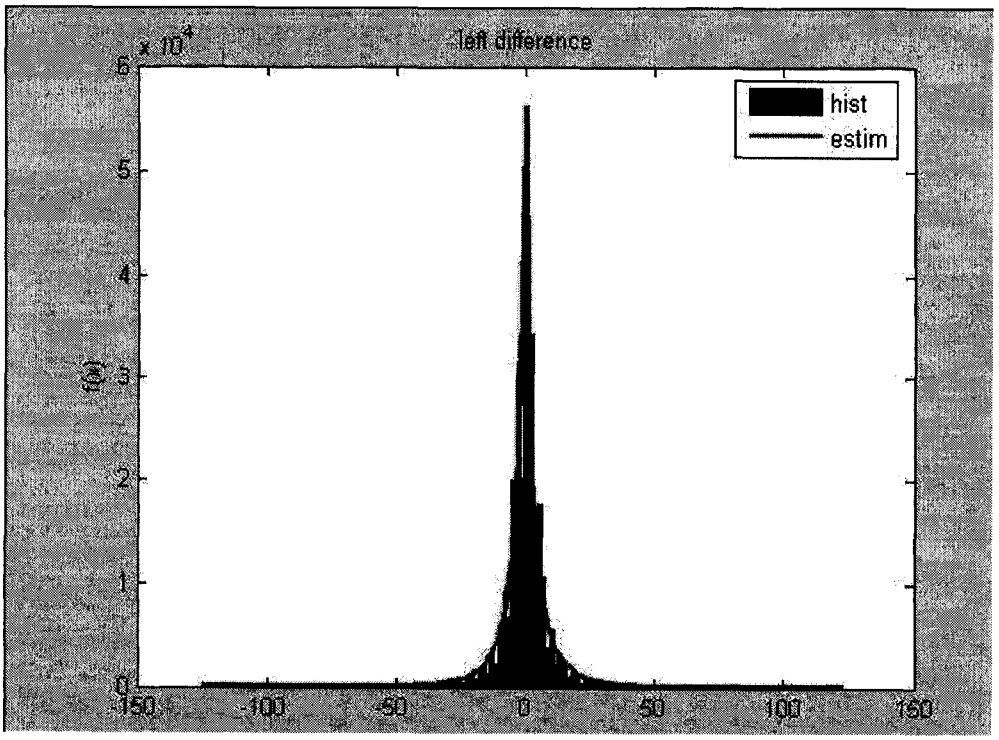

Figure 4.13: The estimation of GGD model on the left difference of Lena image.

is the Lena image which has become the industry standard for judging imaging technology since 1972. Figures 4.13 [25], 4.14 [25], and 4.15 [25] show the fitting of the models of GGD on the distributions of pixel differences for the Lena image. We can see that the models closely fit the distributions.

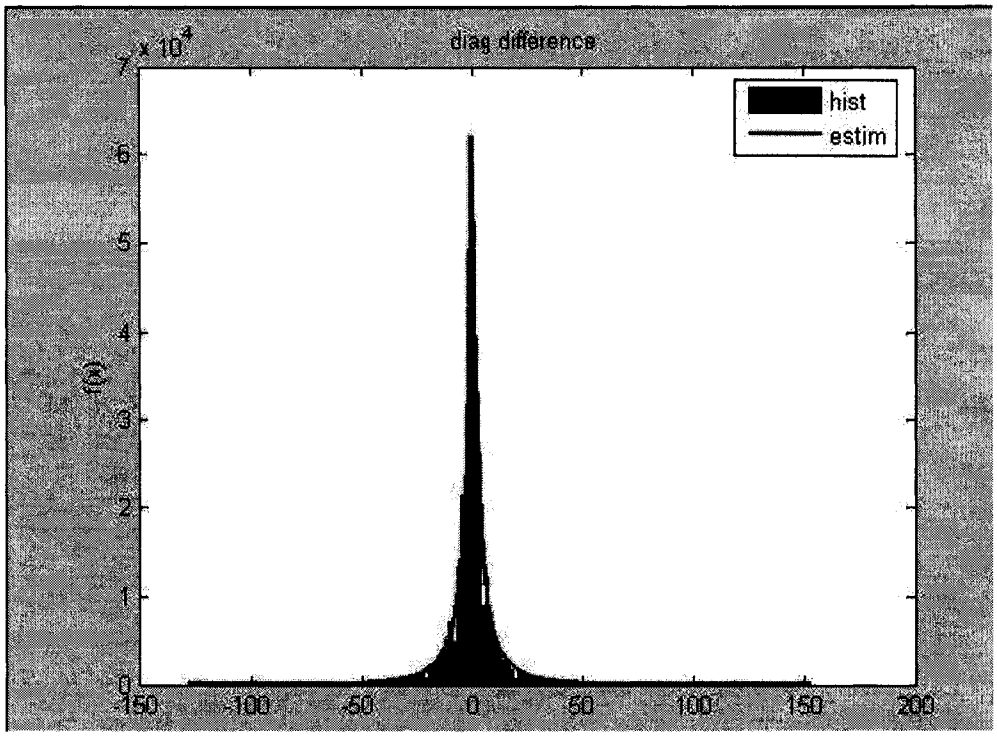

Figure 4.14: The estimation of GGD model on the diagonal differences of Lena image.

Since oil sand image is rich of the texture information, we use GGD to model the statistics of texture operators on oil sand images, in particular, the distribution of pixels difference. The shape parameter and the scale parameter obtained from the model are used as the features for machine learning. Specifically, the segmentation is used as a mask to separate the coarse (object) and fine areas (background) on the input image. Pixels differences are then calculated for the fine and the coarse parts, respectively. The distributions of pixels difference are then modeled by GGD. The shape parameter and the scale parameter obtained from the model are concatenated to form the feature vector for machine learning. Figure 4.16 shows how the feature vector is set up.

To get a complete feature vector to represent the performance of segmentation, the multi resolution decomposition of an image is computed with Gaussian filtering

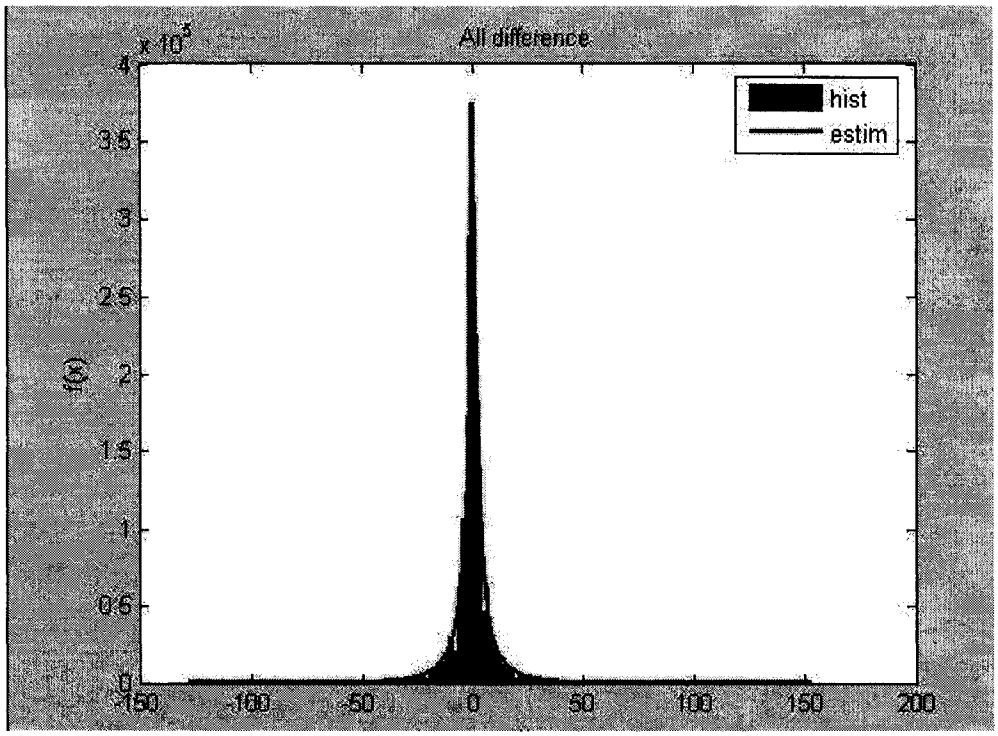

Figure 4.15: The estimation of GGD model on all the differences of Lena image.

and implemented with a pyramid for efficiency. The image at each resolution yields a different pixel differences histogram. The features use α and β from the pixels difference histogram of fine and coarse part on the original image together with α and β from pixel-difference histogram for each resolution.

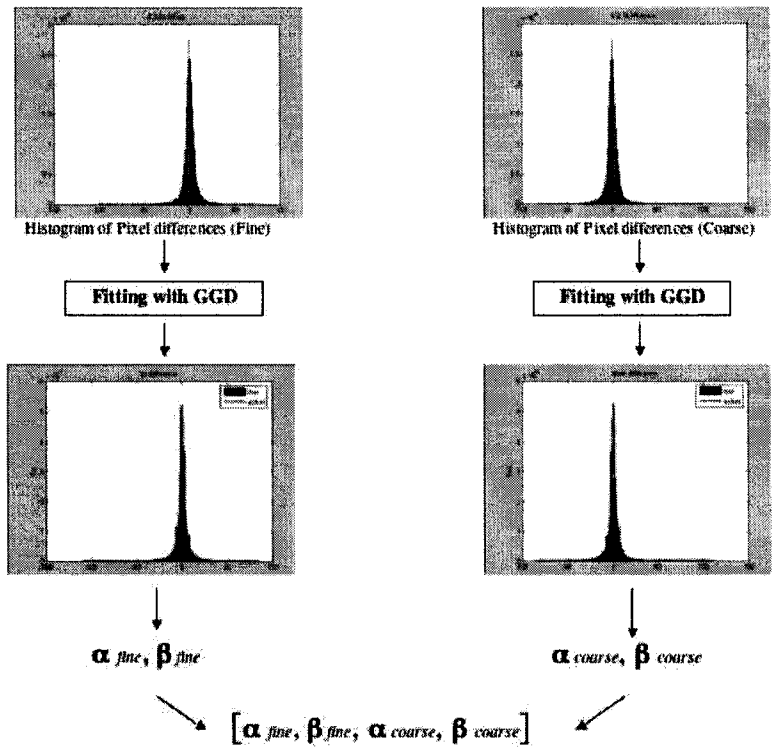

Figure 4.16: Setting up feature vector.

4.2.4 Pruning Strategy: Machine Learned Branch Expansion

The "least-commitment" control policy of MR ADORE is achieved by an exhaustive search of all permutations of operator sequences up to a limited length. Each operator sequence is applied to the input image, resulting in a labeled interpretation. This labeled interpretation is then used to extract features from the original input image. The resulting features are used as input to the machine-learned value function to select the most promising interpretation.

In this work, we investigate a new pruning strategy called the machine learned branch expansion. This pruning strategy eliminates the computationally expensive application of all operator parameter sets. It dynamically discards many parameter

combinations, greatly improving the efficiency of our system while at the same time maintaining the accuracy.

Instead of only one Q-function as in MR ADORE, the new control policy utilizes several Q-functions, one for each of the ordered vision procedures. During the off-line training phase, an intermediate vision procedure, followed by operators with parameters obtained from the best static algorithm, is applied to segment the input image. The reward of the segmentation result is then used as the reward of this vision procedure. A Q-function for this procedure is approximated by training a back-propagation neural network, which takes feature vectors from the intermediate images as input and the rewards as training targets. During the online interpretation phase, the control policy evaluates these Q-functions on the current data and selects the intermediate state that achieves the highest Q-value.

The efficiency of the adaptive OSA can be increased exponentially when the machine learned branch expansion strategy is applied. Suppose the adaptive OSA works with m levels of vision procedures and n possible values for the parameters in each vision procedure. Due to the exhaustive search, the computational complexity of adaptive OSA with the least-commitment policy is O(n^m). However, the computational complexity of the adaptive OSA with the machine learned branch expansion strategy is only O(nxm). Thus the efficiency of the adaptive OSA is increased exponentially when the machine learned branch expansion strategy is applied.

4.3 Summary

We design a system to achieve the adaptive parameter selection. In particular, we adopt MR ADORE as a framework, parameterizing it with the OSA image processing operators, a new scoring metric, a new feature extraction technique based on GGD, and a new pruning strategy called the machine learned branch expansion. The GGD-based features make it possible for our system to machine-learn a dynamic parameter selection policy. The new scoring metric evaluates the segmentation quality more accurately than prior metrics, and thus helps the system learn a better control policy. Finally, the new pruning policy substantially improves the efficiency of the system.

Chapter 5

Experiments

We have presented the design of the adaptive parameter selection system in the last chapter. In this chapter, we present the experiments where the automatic parameter selection system is applied to the oil sand images. The experiment results show that a GGD-based feature is the best among the six we have tested, and that the adaptive parameter selection can not only achieve a better segmentation than the best static algorithm available but also is robust to novel images.

5.1 Performance of Feature Extraction Methods

In this experiment, we set out to compare the effects of six features on the segmentation quality. Each feature extraction method is applied to segmenting the same set of 300 oil sand ore images. Performance data are gathered and compared. Based on the data, GGD (3L) is identified as the best feature extraction method for the segmentation quality and used in subsequent experiments.

The following six features are used in the experiment. GGD (1L) consists of a pair of the shape parameter and the scale parameter from an estimated GGD model of the pixel differences of an image, GGD (2L) concatenates the 2 pairs of the shape and scale parameters from the two GGD models for the 2 levels of a 2-level image pyramid, and GGD (3L) concatenates the 3 pairs of parameters from the 3 GGD models for the 3 levels of a 3-level image pyramid. MRH (1L) applies the intensity histogram of an image as the feature vector, MRH (2L) concatenates the intensity histograms of a 2-level image pyramid, and MRH (3L) concatenates the 3 intensity histograms from a 3-level image pyramid.

Table 5.1 shows the performance of the six features tested. The significance level is set at 0.05. Statistics are based on the differences between the on-line and the off-line reward scores

$$P = \frac{R_{offline} - R_{online}}{R_{offline}} \tag{5.1}$$

	MRH (1L)	MRH (2L)	MRH (3L)	GGD (1L)	GGD (2L)	GGD (3L)
Mean	4.80%	4.62%	4.23%	6.81%	3.74%	2.53%
Variance	0.37%	0.36%	0.43%	0.34%	0.35%	0.34%
Confidence Interval (95% confidence)	4.11% -5.49%	3.94% -5.31%	3.48% -4.97%	6.15% -7.47%	3.06% -4.41%	1.87% -3.20%

Table 5.1: Statistics of performance under different feature extraction methods.

where $R_{offline}$ represents the best reward score obtained for an image using the OSA, and R_{online} represents the reward score from the on-line interpretation stage. Therefore, the smaller the P, the better a feature is.

5.2 Performance of Adaptive Parameter Selection System

We conducted two experiments to compare the performance of the adaptive parameter selection system and the best static OSA, and to compare the Machine Learned Branch Expansion strategy with the least-commitment policy of MR ADORE. Since the GGD (3L) achieves the highest online segmentation quality, the 12 GGD parameters from GGD (3L) are used as features. Cross-validation error is monitored during the training process to guard against over-fitting. The image segmentation system, Ore size Analyst (OSA), is selected as the segmentation algorithm. Two critical parameters of OSA are chosen for adaption, tabulated in regular intervals so that multiple parameter combinations can be applied. The segmentation quality is measured by the fragment based similarity scoring metric.

5.2.1 Experimental Data

Our adaptive parameter selection system has been tested on outdoor images with variations in both the position of the light source and other environmental conditions. Imagery of this type allows us to monitor the system's ability to compensate for real world conditions. As the experiment results will show, the image segmentation system with adaptive parameter selection technique is effective in compensating for the changes observed in these images.

Since we are undertaking a project to obtain the size distribution from oil sand images, in our experiments, we are working with the oil sand images. A total of 150 oil sand images are acquired with three video tapes from different seasons: 50 in April, 50 in August, and 50 in October, all in 2002. The image captured

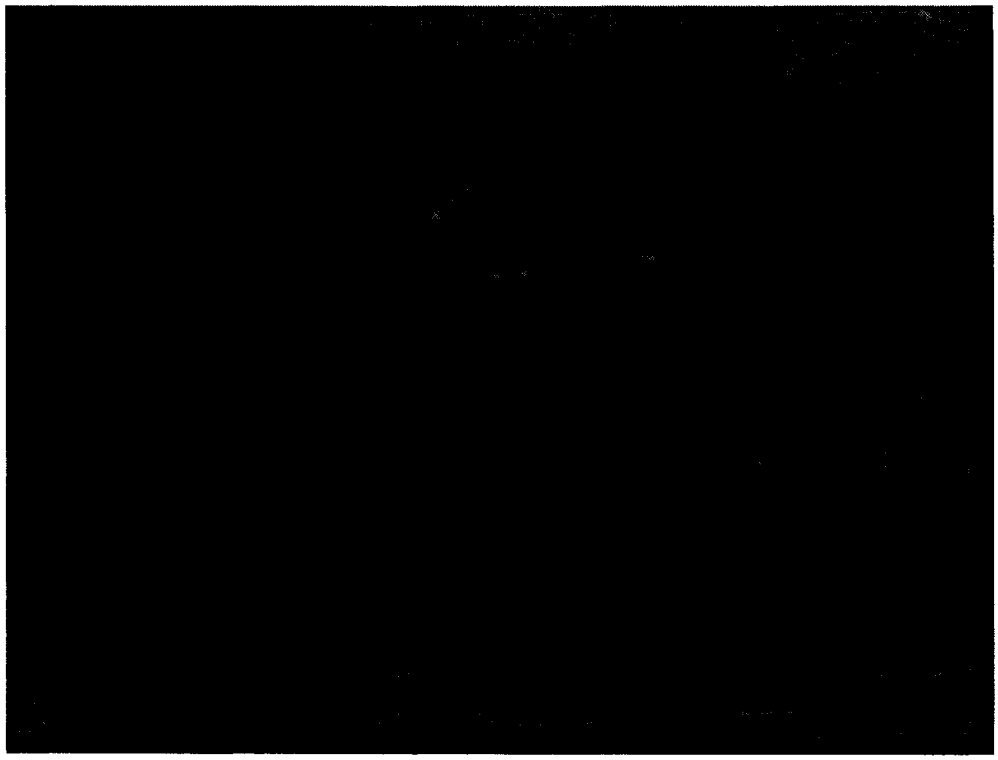

Figure 5.1: A sample image captured from video tape in April.

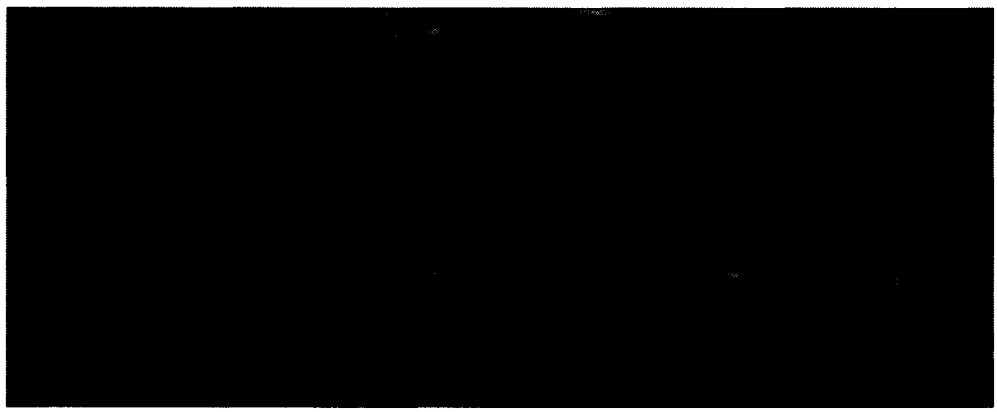

Figure 5.2: A sample image after cropping.

Figure 5.3: A sample ground truth created mannually.

from video tape is 640 by 480 color image as shown in Figure 5.1. To get rid of the irrelavant information like conveyor belt on the image, we have to crop the image to get the interest region. After being cropped, the image size is 500 by 200 as shown in Figure 5.2. Images from different seasons vary in illumination, environmental conditions, and fragment sizes, as shown in Figure 5.4. Images in April and October are used to build the machine learning model, and to select a global optimal parameter set. The trainedmodel is then applied to produce online segmentation results on the remaining 50 images, which are taken in August, but have never been seen by the model.

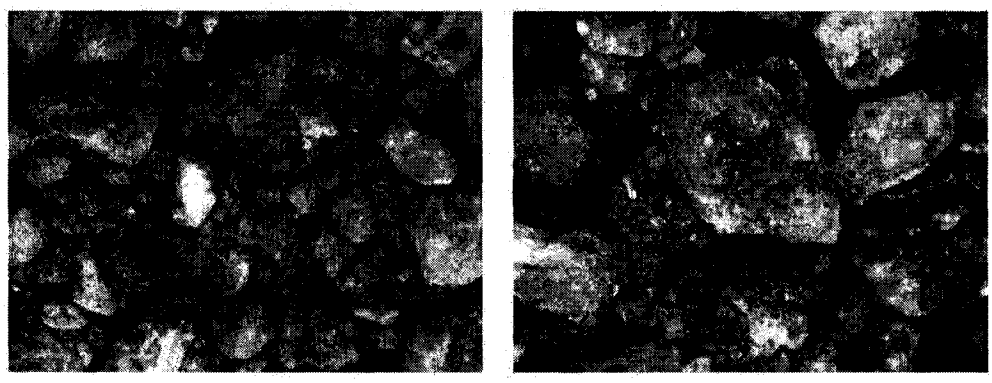

Figure 5.4: Two dissimilar images. The left one is from April, and the right one is from August.

High-quality ground truth images are critical to the accurate evaluation of the segmentation quality. Therefore, we have manually created the ground truths for

150 images and used them in our experiments. Figure 5.3 represents a sample ground truth created mannually for the oil sand image shown in Figure 5.2.

5.2.2 Online Performance: Adaptive Parameter Selection versus Best Static OSA

In this experiment we compare the performance of the adaptive parameter selection system and the best static OSA.

Our system adopts the least-commitment policy with an Artificial Neural Network as the function approximator over a feature vector of 12 GGD parameters. Learning is done in two stages, offline and online. During the offline, all the possible combinations of parameters are generated and applied on the raw input image. The segmented results are then evaluated against the corresponding ground truth. To make the function approximator machine-learned, the descriptive features have to be extracted from the raw image and the segmented result. The neural network takes the features extracted as input vector and the reward score, created by segmentation evaluation, as training signal to learn the mapping from the input images to the corresponding optimal parameter sets. During the online stage, given an unseen raw image, the adaptive segmentation system can adaptively select an optimal parameter set to achieve the best segmentation.

The best static OSA configures the OSA with the globally optimal parameter set, which is obtained by applying the segmentation system OSA with all the possible parameter sets on the training images. An average reward score on the segmentation quality is calculated for each parameter set. The parameter set that achieves the highest average reward score is selected as the globally optimal parameter set.

Figure 5.5 depicts the absolute score difference between the adaptive parameter selection and the best static OSA. The result shows that for 45 out of 50 test images, the adaptive parameter selection produces better segmentation. The improvement for the best case is 18.7%. For 5 out of 50 test images the adaptive parameter selection performs smaller reward scores, and the largest decrease is 5.5%.

Another way to appreciate the performance improvement achieved by the adaptive parameter selection is to use a mean score relative to $R_{offline}$. Our system achieved a mean score of 96% relative to $R_{offline}$, while the best static OSA had a mean score of 87%.

Figure 5.6 is an example that illustrates the improvement the adaptive parameter selection can achieve. It can be seen that the best static OSA mistakenly segments the biggest fragment around the center into many small lumps while the adaptive parameter selection achieves a much better segmentation. It is also worth noting that the scoring metric used penalizes not only every pixel mislabeled but also incorrect splitting and merging of the fragments, and thus tends to produce smaller reward scores than other metrics.

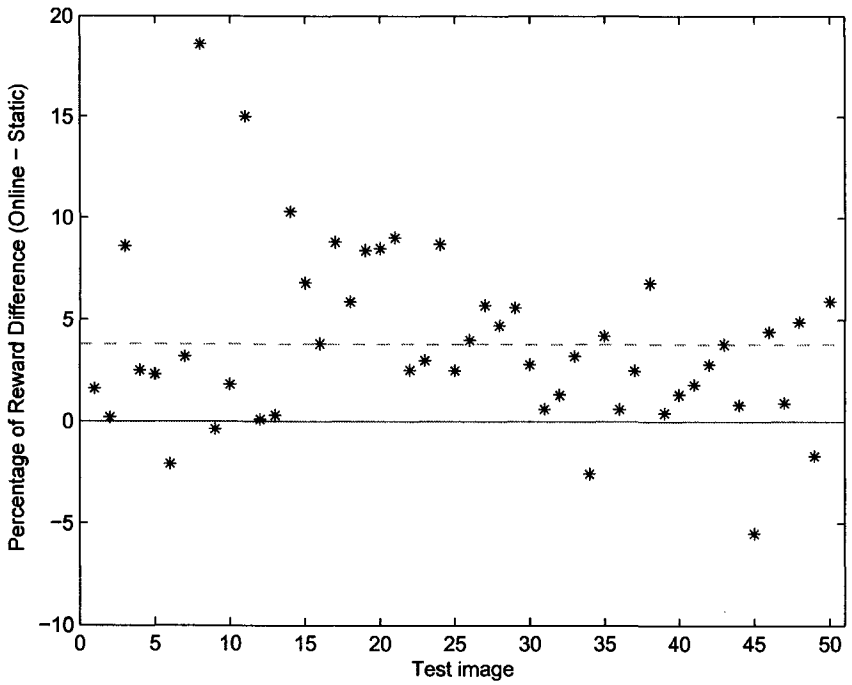

Figure 5.5: Reward score difference between online parameter selection and the best static OSA. The dashed line represents the mean difference.

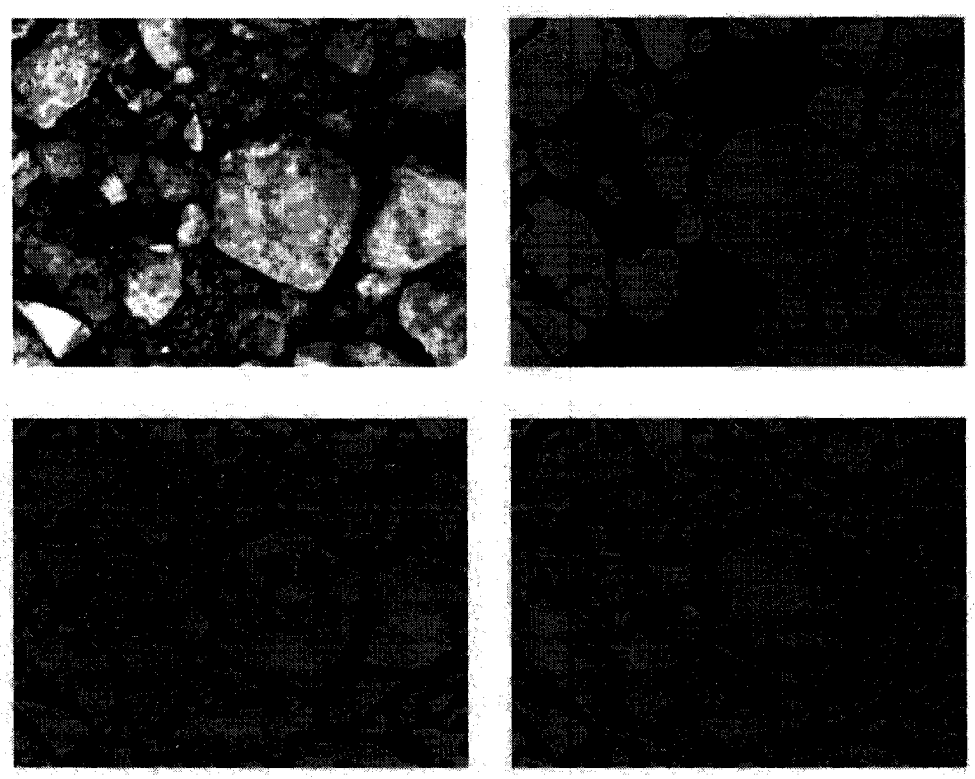

Figure 5.6: Sample showing improvement due to adaptive parameter selection. The upper left is a test oil sand image, and the upper right is its ground truth. The lower left is the segmentation produced by the best static OSA, with a reward score of 0.326. The lower right is the segmentation produced by the adaptive parameter selection, with a reward score of 0.476.

Our learning system is stable. The experiment has been repeated multiple times. Multiple settings, such as the learning rates, momentum, random seeds, and hidden units, are also tried for the learning system. The results show that there is a minor difference of less than 1.7%, on the segmentation performance from the different settings.

5.2.3 Online Performance: Machine Learned Branch Expansion versus Least-Commitment

In the experiments we applied the Machine Learned Branch Expansion Policy with Artificial Neural Networks as the function approximators over feature vectors with 12 GGD parameters. Instead of applying only one Q-function as in the least-commitment policy, the new control policy is built up from a set of Q-functions, one for each vision procedure. The experiment is conducted on the same data set as the experiment reported in the last section.

Learning is also done in two stages, offline and online. During offline, after all the possible parameters are applied on the current data token, new data tokens are produced. New data tokens take several additional vision procedures whose parameters are determined according to the best static algorithm to form the hypothesis. The hypotheses are then evaluated against the corresponding ground truth.

The descriptive features for each vision procedure have to be extracted from the raw image and the hypotheses respectively to make function approximator of each vision procedure machine-learned. Specifically, the hypothesis of each vision procedure is used as a mask to get coarse (object) and fine (background) area information such that the area information can be applied on raw input image to figure out the pixel differences for both fine and coarse part. The distributions of pixel differences are then modeled by GGD respectively to obtain the shape and scale parameters to describe fine and coarse part for each vision procedure.

To get a complete feature vector to represent the performance, for each vision procedure the multi resolution decompositions of images (raw image and hypothesis) are computed with Gaussian filtering and implemented with a pyramid (n=3) for efficiency. α and β from estimations of GGD model on the pixel differences histogram of fine and coarse part of each resolution are concatenated to set up the feature vector.

A neural network function is trained for each vision procedure by taking the features extracted as input vector and the reward score as training signal to learn the mapping from the intermediate state to the corresponding optimal parameter.

During the online stage, once given a unseen raw image, the control policy which is learned can then be applied to adaptively select the optimal parameters for every vision procedure, in particular, select one of the intermediate states which achieves the highest reward score to continue the segmentation process for every

	Least-Commitment	MLBE	Difference (MLBE - LC)
Image 51	0.464	0.523	5.9%
Image 52	0.587	0.591	0.4%
Image 53	0.427	0.426	- 0.1%
Image 54	0.555	0.571	1.6%
Image 55	0.523	0.530	0.7%
Image 56	0.513	0.498	- 1.5%
Image 57	0.559	0.559	0
Image 58	0.617	0.615	- 0.2%
Image 59	0.389	0.389	0
Image 60	0.511	0.511	0

Table 5.2: Online performance under least commitment and machine learned branch expansion. Numbers represent the absolute reward score. Models are built with 100 images.

vision procedure, such that the best segmentation can be obtained based on the segmentation algorithm.

Table 5.2 depicts the score differences between the performance of the least commitment policy and the machine learned branch expansion (MBLE) for 10 images. MBLE achieves a mean score of 52.1%, while the least commitment a mean score of 51.4%. In 7 out of the 10 images, the machine learned branch expansion improves over or maintains the same segmentation performance as the least-commitment policy. The improvement for the best case is 5.9%. For 3 out of the 10 images where MBLE decreases the reward score, the decrease in scores is at most 1.5%.

5.3 Summary

We test our adaptive parameter selection system using 150 oil sand images from three seasons. The ground truths for these 150 images are obtained manually. In particular, we empirically select GGD (3L) as the best feature for the machine-learning, and demonstrate that our adaptive system does improve performance over the best static OSA. We also show that an alternative control policy, the machine learned branch expansion, can improve not only efficiency but also segmentation quality over the least-commitment policy.

These preliminary results demonstrate that our approach is robust to novel images and that the adaptive parameter selection can be effective in maximizing the segmentation quality on a per image basis.

Chapter 6

Summary and Future Work

6.1 Summary of Contributions and Results

An image segmentation system with fixed parameters cannot achieve the best segmentation on a per image basis because no a fixed set of optimal parameters exists for all images. Thus an intelligent domain-specific policy is required to control the application of operators and parameters and maximize the segmentation quality.

In this thesis, we design a system to achieve the adaptive parameter selection. In particular, we adopt MR ADORE as a framework, parameterizing it with the OSA image processing operators, a new scoring metric, a new feature extraction technique based on GGD, and a new pruning strategy called the machine learned branch expansion. The GGD-based features make it possible for our system to machine-learn a dynamic parameter selection policy. The new scoring metric evaluates the segmentation quality more accurately than prior metrics, and thus helps the system learn a better control policy. Finally, the new pruning policy substantially improves the efficiency of the system. Experiments on oil sand images show that the new system achieves better accuracy than the best static image segmentation algorithm, which takes a set of globally optimal parameters.

6.2 Future Work

The initial success suggests some improvements to the system. In particular, we are considering more adaptive strategies to overcome problems caused by non-uniform images. As shown in Figure 6.1, oil sand images tend to contain fragments of mixed sizes. When fragment sizes are different, a fixed window size for the contrast enhancement cannot achieve optimal results. There is always a conflict between big and small fragments in terms of the segmentation quality: while a large window size benefits big fragments, it performs poorly on small fragments, and vice versa for a small window size. Clearly, we need a more effective strategy to solve this conflict

Figure 6.1: A sample oil sand image with fragments of different sizes.

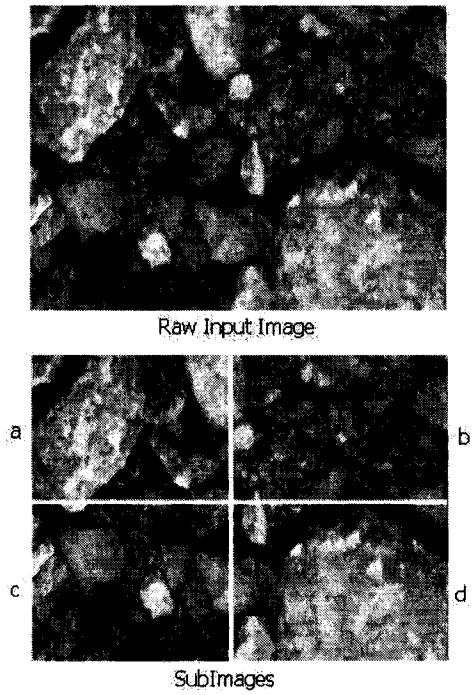

Figure 6.2: Sample subImages after a possible splitting.

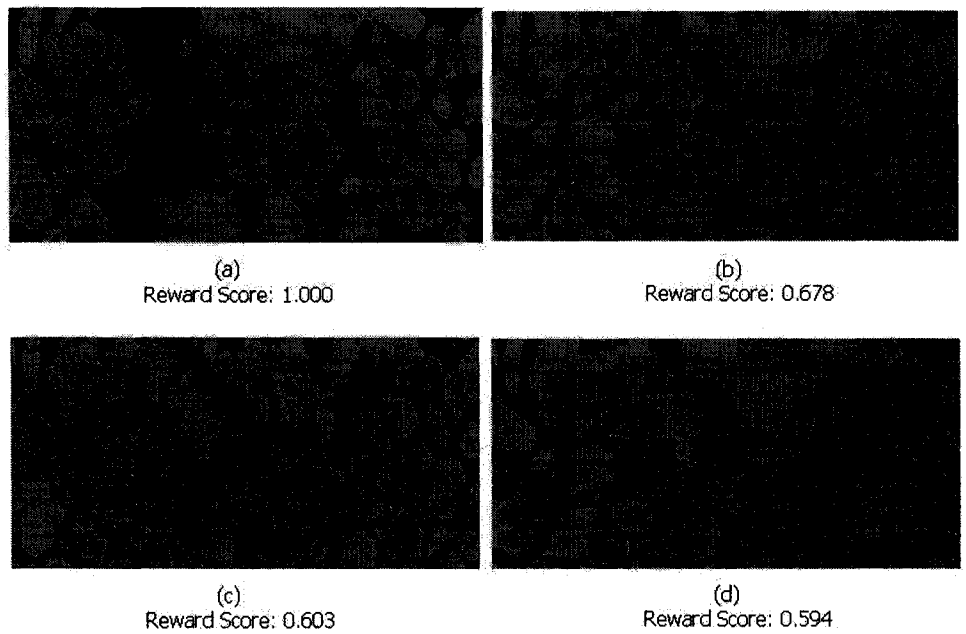

Figure 6.3: Segmentation results on the same raw input image by applying splitting strategy with different processing windows. (a) is the ground truth; (b) and (c) are segmentation results with different splitting process; (d) is a segmentation without splitting process;

in order to achieve better segmentation quality. In the following, we outline two possible strategies.

One strategy is to split an input image into a number of reasonable processing windows, as shown in Figure 6.2. Each window is segmented independently. The results are then merged together to form the final segmentation for the whole image. Some experiments have been done with this strategy, and the preliminary empirical results show that image splitting can be an effective method to improve the segmentation quality (see Figure 6.3.)

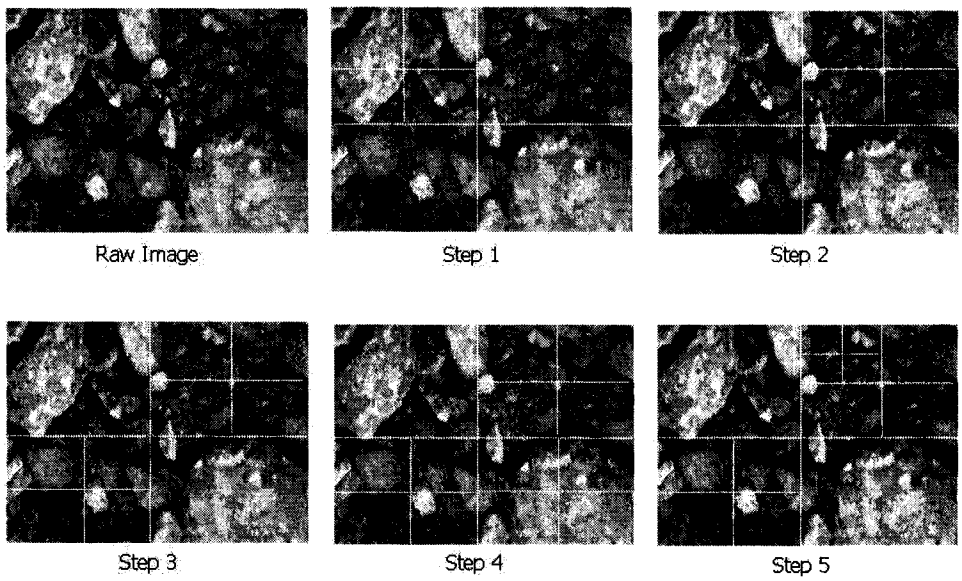

Figure 6.4: Adaptive window size combination with quad-tree separating strategy.

The other strategy is to select the window size adaptively. During the image processing process, if different window sizes can be adopted for different parts of the input image, both segmentation quality and processing time can be improved dramatically. For example, in the contrast enhancement stage, the window size is a key parameter. Currently, we are investigating the adaptive window size combination with the quad-tree separating strategy, as shown in Figure 6.4.

Bibliography

[1] A.P.Mendonca and E.A.B.da Silva. Segmentation approach using local image statistics. *Electronics Letters*, 36(14):1199 – 1201, July 2000.

[2] J. Bach, C. Fuler, A. Gupta, A. Hampapur, B. Horowitz, R.Humphrey, R. Jain, and C. Shu. The virage image search engine:an open framework for image management. In *Proc. SPIE Conf.Storage and Retrieval for Image and Video Databases IV*, volume 2670, pages 76–87, 1996.

[3] A.G. Barto, R.S. Sutton, and C.J.C.H. Watkins. Learning and sequential decision making. Technical report, Dept. of Computer and Information Science, Univ. of Mass., Amherst, Mass., 1989.

[4] B. Bhanu. Automatic target recognition: State of the art survey. *IEEE Trans. Aerosp. Electron Syst.*, AES-22:364–379, 1986.

[5] B. Bhanu, S. Lee, and S. Das. Adaptive image segmentation using genetic and hybrid search method. *IEEE Transactions on Aerospace and Electronic Systems*, 31:1268–1291, 1995.

[6] B. Bhanu, S. Lee, and J. Ming. Adaptive image segmentation using a genetic algorithm. *IEEE Transactions on Systems, Man, and Cybernetocs*, 25:1543–1568, 1995.

[7] B. Bhanu and J. Ming. Recognition of occluded objects: A cluster-structure algorithm. *Pattern Recognition*, 20:199–211, 1987.

[8] B. Bhanu and J. Peng. Adaptive integrated image segmentation and object recognition. *IEEE Transactions on Systems, Man, and Cybernetocs - Part C: Application And Reviews*, 30:427–441, 2000.

[9] R. Buccigrogrossi and E. Simoncelli. Image compression via joint statistical characterization in the wavelet domain. *IEEE Trans. Image Proc.*, 8(12):1688–1701, 1999.

[10] B. Draper, U. Ahlrichs, and D. Paulus. Adapting object recognition across domains: A demonstration. In *Proceedings of International Conference on Vision Systems*, pages 256–267, Vancouver, B.C., 2001.

[11] B. Draper, J. Bins, and K. Baek. Adore: adaptive object recognition. *Videre*, 1:86–99, 2000.

[12] B. Funt and G. Finlayson. Color constant color indexing. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 17:522–529, 1995.

[13] D. Geman and B.Jedynak. An active testing model for tracking roads in satellite images. *IEEE Trans. PAMI*, 18(1):1–114, 1996.

[14] D Goldberg. *Genetic algorithms in search, optimization, and machine learning*. Addison-Westley, 1987.

[15] E. Hadjidemetriou, M. Grossberg, and S. Nayar. Spatial information in multiresolution histograms. In *Proc. Computer Vision and Pattern Recognition Conf.*, volume 1, pages 702–709, 2001.

[16] E. Hadjidemetriou, Michael D. Grossberg, and Shree K. Nayar. Multiresolution histograms and their use for recognition. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 26:831–847, 2004.

[17] J. Koenderink. The structure of images. *Biological Cybernetics*, 50:363–370, 1984.

[18] K. I. Laws. The phoenix image segmentation system: Description and evaluation. Technical Report 289, SRI International Technical Note, 1982.

[19] I. Levner, V. Bulitko, L. Li, G. Lee, and R. Greiner. Towards automated creation of image interpretation systems. In *Gedeon, T., and Fung, L., eds., Lecture Notes in Artificial Intelligence*, volume 2903, pages 653–665, Berlin, 2003.

[20] I. Levner. Multi resolution adaptive object recognition system: A step towards autonomous vision sustem. Technical report, University of Alberta, 2003.

[21] I. Levner and V. Bulitko. Machine learning for adaptive image interpretation. In *Proceedings of the National Conference on Artificial Intelligence (AAAI) and Innovative Applications of Artificial Intelligence Conference (IAAI)*, pages 870 – 876, San Jose, California, 2004.

[22] W. Niblack. The qbic project: Querying images by content using color, texture, and shape. In *Proc. SPIE Conf. Storage and Retrieval for Image and Video Databases*, volume 1908, pages 173–187, 1993.

[23] J. Peng and B. Bhanu. Closed-loop object recognition using reinforcement learning. *IEEE Trans. PAMI*, 20:139–154, 1998.

[24] J. Peng and B. Bhanu. Delayed reinforcement learning for adaptive image segmentation and feature extraction. *IEEE Trans. SMC*, 28:482–488, 1998.

[25] R. KrupiDski. The pixels difference of the Lena image: the estimation of GGD model.

[26] H. Sidenbladh and M. Black. Learning image statistics for bayesian tracking. In *Proceedings of IEEE 2001 International Conference on Computer Vision*, volume II, pages 709–716, 2001.

[27] M. Stricker and M. Orengo. Similarity of color images. In *Proc.SPIE Conf. Storage and Retrieval for Image and Video Databases III*, volume 2420, pages 381–392, 1995.

[28] R. Sutton and A Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2000.

[29] M. Swain and D. Ballard. Color indexing. *Int l J. Computer Vision*, 7:11–32, 1991.

[30] X. Wang. A survey of image statistics and applications. Technical report, Department of Computing Science, University of Alberta, 2004.

[31] X. Wang, M. Polak, V. Bulitko, and H. Zhang. Machine learning for adaptive parameter selection in ore image segmentation. In *Proceedings of the National Conference on Artificial Intelligence (AAAI), Workshop on Learning in Computer Vision*, Pittsburgh, Pennsylvania, 2005.

[32] C. Watkins. *Learning from Delayed Rewards*. PhD thesis, Kings College, Cambridge University, 1989.

[33] A. Witkin. Scale-space filtering. In *Proc. of Int l Joint Conf. of Artificial Intelligence*, pages 1019–1022, 1983.

[34] H. Zhang, A. Kankanhali, and S. Smoliar. Automatic partitioning of full-motion video. *Multimedia Systems*, 1:10–28, 1993.

[35] H. Zhang, C. Low, W. Smoliar, and J. Wu. Video parsing, retrieval and browsing: An integrated and content-based solution. *ACM Multimedia*, pages 15–24, 1995.

[36] S. Zhu and D. Mumford. Prior learning and gibb reaction-diffusion. *IEEE Trans. PAMI*, 19(11):1236–1250, 1997.

Appendix A

Reward Scores from Off-line Training Stage of Adaptive OSA

Table A.1, Table A.2, Table A.3, Table A.4, and Table A.5 show the best possible scores (absolute) obtained from Off-line training stage of Adaptive OSA for each image on experiments . The score range is [0, 1]. Totally 150 oil sand ore images were applied on the experiments.

Image#	Reword Score (Absolute)
1	0.495
2	0.518
3	0.57
4	0.575
5	0.503
6	0.621
7	0.528
8	0.643
9	0.429
10	0.601
11	0.629
12	0.502
13	0.453
14	0.461
15	0.535
16	0.43
17	0.478
18	0.574
19	0.578
20	0.531
21	0.551
22	0.531
23	0.584
24	0.478
25	0.419
26	0.451
27	0.406
28	0.463
29	0.468
30	0.435

Table A.1: The best possible scores (absolute) obtained from Off-line training stage of Adaptive OSA for each image on experiments . The score range is [0, 1].

Image#	Reword Score (Absolute)
31	0.624
32	0.52
33	0.534
34	0.512
35	0.586
36	0.514
37	0.529
38	0.61
39	0.553
40	0.514
41	0.513
42	0.453
43	0.44
44	0.555
45	0.575
46	0.452
47	0.472
48	0.459
49	0.55
50	0.527
51	0.429
52	0.498
53	0.345
54	0.464
55	0.438
56	0.461
57	0.441
58	0.572
59	0.363
60	0.486

Table A.2: The best possible scores (absolute) obtained from Off-line training stage of Adaptive OSA for each image on experiments . The score range is [0, 1].

Image#	Reword Score (Absolute)
61	0.476
62	0.548
63	0.455
64	0.442
65	0.279
66	0.389
67	0.447
68	0.41
69	0.457
70	0.512
71	0.392
72	0.531
73	0.403
74	0.414
75	0.535
76	0.456
77	0.455
78	0.432
79	0.38
80	0.42
81	0.446
82	0.376
83	0.406
84	0.441
85	0.384
86	0.549
87	0.396
88	0.561
89	0.503
90	0.528

Table A.3: The best possible scores (absolute) obtained from Off-line training stage of Adaptive OSA for each image on experiments . The score range is [0, 1].

Image#	Reword Score (Absolute)
91	0.525
92	0.473
93	0.498
94	0.506
95	0.508
96	0.434
97	0.518
98	0.438
99	0.401
100	0.488
101	0.511
102	0.504
103	0.495
104	0.578
105	0.413
106	0.474
107	0.488
108	0.446
109	0.502
110	0.476
111	0.488
112	0.445
113	0.557
114	0.538
115	0.585
116	0.494
117	0.456
118	0.51
119	0.646
120	0.409

Table A.4: The best possible scores (absolute) obtained from Off-line training stage of Adaptive OSA for each image on experiments . The score range is [0, 1].

Image#	Reword Score (Absolute)
121	0.51
122	0.41
123	0.561
124	0.324
125	0.533
126	0.439
127	0.455
128	0.482
129	0.517
130	0.536
131	0.448
132	0.489
133	0.516
134	0.447
135	0.534
136	0.454
137	0.562
138	0.486
139	0.485
140	0.514
141	0.502
142	0.432
143	0.502
144	0.547
145	0.534
146	0.489
147	0.512
148	0.641
149	0.42
150	0.555

Table A.5: The best possible scores (absolute) obtained from Off-line training stage of Adaptive OSA for each image on experiments . The score range is [0, 1].

Appendix B

Absolute Scores of Testing Images from Off-line, Online, and the Best Static OSA

Table B.1 and Table B.2 show the absolute Scores of testing images from Off-line (training stage of Adaptive OSA), Online (evaluation stage of Adaptive OSA), and the best static OSA . The score range is [0, 1]. Totally 50 oil sand ore images were involved on the testing.

Image#	Score (Offline)	Score (Online)	Score (the Best Static OSA)
51	0.429	0.419	0.403
52	0.498	0.438	0.436
53	0.345	0.337	0.251
54	0.464	0.446	0.421
55	0.438	0.438	0.415
56	0.461	0.398	0.419
57	0.441	0.434	0.402
58	0.572	0.523	0.337
59	0.363	0.33	0.334
60	0.486	0.463	0.445
61	0.476	0.476	0.326
62	0.548	0.53	0.529
63	0.455	0.445	0.442
64	0.442	0.442	0.339
65	0.279	0.279	0.211
66	0.389	0.379	0.341
67	0.447	0.447	0.359
68	0.41	0.409	0.35
69	0.457	0.457	0.373
70	0.512	0.505	0.42
71	0.392	0.392	0.302
72	0.531	0.519	0.494
73	0.403	0.403	0.373
74	0.414	0.388	0.301
75	0.535	0.535	0.51
76	0.456	0.415	0.375
77	0.455	0.455	0.398
78	0.432	0.417	0.37
79	0.38	0.38	0.324
80	0.42	0.418	0.39

Table B.1: The absolute Scores of testing images from Off-line (training stage of Adaptive OSA), Online (evaluation stage of Adaptive OSA), and the best static OSA . The score range is [0, 1].

Image#	Score (Offline)	Score (Online)	Score (the Best Static OSA)
81	0.446	0.413	0.407
82	0.376	0.294	0.281
83	0.406	0.318	0.286
84	0.441	0.372	0.398
85	0.384	0.384	0.342
86	0.549	0.544	0.538
87	0.396	0.394	0.369
88	0.561	0.556	0.488
89	0.503	0.477	0.473
90	0.528	0.518	0.505
91	0.525	0.519	0.501
92	0.473	0.442	0.414
93	0.498	0.465	0.427
94	0.506	0.45	0.442
95	0.508	0.446	0.501
96	0.434	0.434	0.39
97	0.518	0.512	0.503
98	0.438	0.428	0.379
99	0.401	0.365	0.382
100	0.488	0.482	0.423

Table B.2: The absolute Scores of testing images from Off-line (training stage of Adaptive OSA), Online (evaluation stage of Adaptive OSA), and the best static OSA . The score range is [0, 1].

Appendix C

Relative Scores of Testing Images from Online and the Best Static OSA

Table C.1 and Table C.2 show the relative reward scores obtained from the best static OSA and online evaluation stage of Adaptive OSA with respect to the best possible scores obtained from offline training stage of Adaptive OSA. The score range is [0, 1].

Image#	RelativeScore (Best Static OSA)	RelativeScore (Online)
51	0.939	0.976
52	0.875	0.879
53	0.727	0.976
54	0.907	0.961
55	0.947	1.000
56	0.908	0.863
57	0.911	0.984
58	0.589	0.914
59	0.920	0.909
60	0.915	0.952
61	0.684	1.000
62	0.965	0.967
63	0.971	0.978
64	0.766	1.000
65	0.756	1.000
66	0.876	0.974
67	0.803	1.000
68	0.853	0.997
69	0.816	1.000
70	0.820	0.986
71	0.770	1.000
72	0.930	0.977
73	0.925	1.000
74	0.727	0.937
75	0.953	1.000
76	0.822	0.910
77	0.874	1.000
78	0.856	0.965
79	0.852	1.000
80	0.928	0.995

Table C.1: The relative reward scores obtained from the best static OSA and online evaluation stage of Adaptive OSA with respect to the best possible scores obtained from offline training stage of Adaptive OSA. The score range is [0, 1].

Image#	RelativeScore (Best Static OSA)	RelativeScore (Online)
81	0.912	0.926
82	0.747	0.781
83	0.704	0.783
84	0.902	0.843
85	0.890	1.000
86	0.979	0.990
87	0.931	0.994
88	0.869	0.991
89	0.940	0.948
90	0.956	0.981
91	0.954	0.988
92	0.875	0.934
93	0.857	0.933
94	0.873	0.889
95	0.986	0.877
96	0.898	1.000
97	0.971	0.988
98	0.865	0.977
99	0.952	0.910
100	0.866	0.987

Table C.2: The relative reward scores obtained from the best static OSA and online evaluation stage of Adaptive OSA with respect to the best possible scores obtained from offline training stage of Adaptive OSA. The score range is [0, 1].

Appendix D

Score Differences between Scores Obtained from Online and the Best Static OSA

Table D.1 and Table D.2 show the differences between the scores obtained from the online evaluation stage of Adaptive OSA and the Best Static OSA (Online - Static). Relative differences represent the score differences relative to the possible best scores from offline. The score range is [0, 1].

Image#	ScoreDiff (Absolute)	ScoreDiff (Relative)
51	0.016	0.039
52	0.002	0.004
53	0.086	0.342
54	0.025	0.059
55	0.023	0.055
56	-0.021	-0.050
57	0.032	0.079
58	0.186	0.551
59	-0.003	-0.011
60	0.018	0.040
61	0.15	0.460
62	0.001	0.002
63	0.002	0.006
64	0.103	0.303
65	0.068	0.322
66	0.038	0.111
67	0.088	0.245
68	0.059	0.168
69	0.084	0.225
70	0.085	0.202
71	0.090	0.298
72	0.025	0.050
73	0.030	0.080
74	0.087	0.289
75	0.025	0.049
76	0.040	0.106
77	0.057	0.143
78	0.047	0.127
79	0.056	0.172
80	0.028	0.071

Table D.1: The differences between the scores obtained from the online evaluation stage of Adaptive OSA and the Best Static OSA (Online - Static). Relative differences represent the score differences relative to the possible best scores from offline.

Image#	ScoreDiff (Absolute)	ScoreDiff (Relative)
81	0.005	0.014
82	0.013	0.046
83	0.032	0.111
84	-0.026	-0.065
85	0.042	0.122
86	0.006	0.011
87	0.025	0.067
88	0.068	0.139
89	0.004	0.008
90	0.013	0.025
91	0.018	0.035
92	0.028	0.067
93	0.038	0.088
94	0.007	0.018
95	-0.055	-0.109
96	0.044	0.11
97	0.009	0.017
98	0.049	0.129
99	-0.017	-0.044
100	0.059	0.139

Table D.2: The differences between the scores obtained from the online evaluation stage of Adaptive OSA and the Best Static OSA (Online - Static). Relative differences represent the score differences relative to the possible best scores from offline.