Is "the world we see around us {is} the real world itself, or" is it "merely a copy of the world presented to consciousness by our brain in response to input from our senses"?

– From "The World in Your Head" by Steve Lehar [23].

University of Alberta

USING MULTIPLE GESTALT FEATURES FOR LARGE OIL SAND LUMP DETECTION

by

Yury Potapovich ©

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
Spring 2008

Your file Votre référence
ISBN: 978-0-494-45872-3
Our file Notre référence
ISBN: 978-0-494-45872-3

Abstract

A new multiple feature-based method of object detection in sequences of digital images is introduced. The proposed method, unlike most existing feature-based computer vision methods, uses features that correspond to perceptual grouping principles of the Gestalt theory of psychology: similarity, common motion and goodness of shape. Gestalt features have been shown to be perceptually significant and directly related to geometric structure of real world scenes. There has been only a limited amount of research done on using multiple Gestalt features, which are important for the detection of salient objects. The proposed method applies Gestalt ideas to a computer vision problem (detection of large oil sand lumps) and confirms that using multiple gestalt features improves the object detection performance compared to using a smaller number of features. The new method employs decision trees to address the problem of a partial gestalt collaboration and conflict (*i.e.,* the problem of feature fusion), which is still a subject of ongoing computer vision research. The proposed method could also be generalized to other object recognition problems, since it uses universal grouping principles of Gestalt theory integrally with automated machine learning decision-making.

Acknowledgements

I sincerely thank my research supervisors Dr. Ron Kube and Dr. Hong Zhang for their advice, mentorship and support throughout my thesis work. I would also like to thank them for my invaluable training in critical thinking, analysis and technical writing. I am especially thankful to Dr. Ron Kube, who encouraged me to continue my studies at the graduate level, which resulted in this thesis.

I am very thankful to the members of my committee, Dr. Hong Zhang, Dr. Ron Kube, Dr. Christopher Sturdy, and Dr. Walter Bischof for their time to review and comment on my thesis.

I also thank NSERC, Syncrude Canada Ltd. and the Centre for Intelligent Mining Systems for the financial support of my research.

Finally, I am extremely grateful to my wife Diana for her support, patience and understanding.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

Abbreviation	Description
CIMS	Centre for Intelligent Mining Systems
GDI	Gini's diversity index
GUI	graphical user interface
MDR	maximum deviance reduction
SSD	sum of squared differences
TP	true positive
TN	true negative
FP	false positive
FN	false negative
i	intensity Gestalt feature
m	motion Gestalt feature
s	shape Gestalt feature
i+m	a combination of intensity and motion Gestalt features
i+s	a combination of intensity and shape Gestalt features
m+s	a combination of motion and shape Gestalt features
i+m+s	a combination of intensity, motion and shape Gestalt features

List of Equations

Chapter 1

Introduction

One of the main goals of computer vision is to create artificial systems that are equal to or surpass human biological vision. A central philosophical issue of biological vision is "the question whether the world we see around us is the real world itself, or whether it is merely a copy of the world presented to consciousness by our brain in response to input from our senses" ([23], p.1). Computer vision ideas seem to be formulated in a latter way. The Gestalt[1] theory also does a very good job describing how sensory input creates various perceptions from the perceptual psychology point of view. So far, Gestalt ideas showed a great potential to be successfully used to solve computer vision problems.

1.1 Motivation

Object recognition and segmentation of digital images is a challenging area of computer vision research. It turns out that another discipline, in addition to computer vision, is also studying how the objects are formed from their parts. The discipline is the Gestalt theory of psychology. Gestaltists describe a fixed set of principles (*i.e.,* partial gestalts[2]) that govern object perception (*i.e.,* creation of a global gestalt). An important property of Gestalt principles is that they reflect the geometrical structure of the real world, which should be useful while detecting/recognizing

[1] A gestalt is defined by [19] as "a structure, configuration, or pattern of physical, biological, or psychological phenomena so integrated as to constitute a functional unit with properties not derivable by summation of its parts."

[2] In this thesis, the word "gestalt" is capitalized when it implies an expression "Gestalt theory" and use lower case when it implies an object or a Gestalt principle.

objects in digital images of the real scenes [27, 10]. Many computer vision approaches do not use more than one or two partial gestalts. However, Desolneux *et al.* ([10], p.20) state that "most salient objects of groups come to sight by several grouping laws". Therefore, it needs to be shown that using multiple features based on Gestalt principles has more potential than using just one or two partial gestalts. This idea is going to be tested on an object recognition task: detection of large oil sand lumps.

Two definitions of partial gestalts are used: according to the first definition a partial gestalt corresponds to the Gestalt principle of perceptual organization, and according to the second one a partial gestalt is the implementation of a corresponding Gestalt principle or, in computer vision terminology, a *feature*. Thus, a *Gestalt feature* is a Gestalt law/principle of pereceptual organization adopted and used in the proposed method (see Section 2.3.3 for more information on correspondences between Gestalt laws and features in the proposed method).

1.2 Application Domain

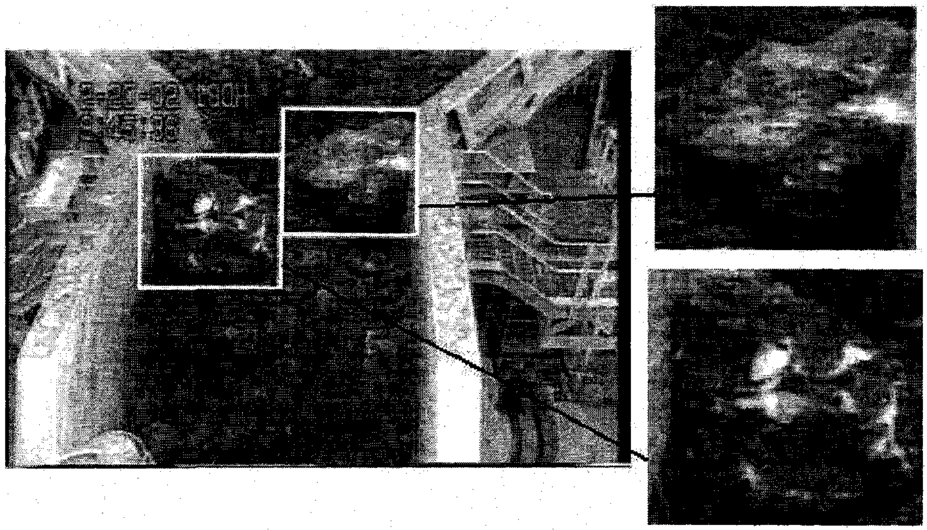


Figure 1.1: A sample image containing two large lumps. Notice uneven texture and brightness within and between the lumps.

One of the most ambiguous pieces of digital image data that have been seen to date are images of oil sand. Oil sand particles may have high variability of texture and brightness within and between themselves (see Figure 1.1 for a sample image). The general quality of images is poor, since the production line is situated outdoors and is subject to changing weather and lighting conditions. The appearance of oil sand and the quality of the image data make oil sand segmentation a very complex task even for humans.

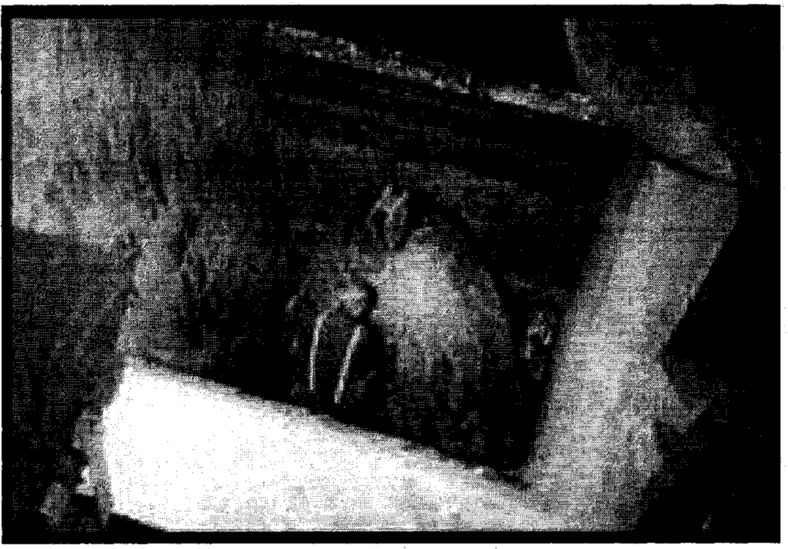


Figure 1.2: A large lump, which jammed the crushing system (double-roll crusher). A worker can be seen next to the large lump that is on top of a crusher.

Detection of large lumps of oil sand is also an important production issue for Syncrude Canada Ltd. Some large lumps can block the main production lines, which requires manual removal of lumps causing significant production downtime [12]. This typically happens during winter, when large frozen lumps fall into the oil sand crushing mechanism (see Figure 1.2 for an example). Timely detection of oversized lumps would allow the plant operators to avoid crusher jams by stopping the conveyor before the lump reaches the crusher.

The problem of large lump detection can be viewed as an object recognition via segmentation problem, where large lump objects have to be separated from the rest of the oil sand. Large lump detection was chosen to be the application domain of this thesis, since it is both a complex and a useful segmentation task.

1.3 Method Structure

Image edges were chosen as the basic input elements for the proposed method, because edges are a Gestalt quality as outlined by Desolenux *et al.* in [6, 9]. The definition of a gradient edge seems to follow the law of connectedness,[3] where gradient pixels of higher values are grouped together. Edges are an important general application-specific feature based on observations of large lump image data.

The proposed method exhaustively goes through all possible combinations of edges in each digital image. Edges in each combination are then connected to each other forming a contour (this way the partial gestalt of closure is imitated). *Thus, there is a number of candidate contours for each image and the main task becomes grouping these candidates in two sets: large lumps and non-large (i.e., smaller) lumps.* The grouping is performed using Gestalt features (*i.e.,* partial gestalts). The following partial gestalts were chosen to be used in the proposed method: similarity, common motion and goodness of shape. Hence, candidates are assigned into the large lump set (*i.e.,* detected as large lumps) by their intensity similarity, velocity and their shape characteristics.

The collaboration of partial gestalts is a major problem in related research and is still under investigation [10, 4]. In the proposed work the collaboration of Gestalt features is provided by using *a priori* information about their relationships, which is embodied into decision trees. So, a decision tree is trained on a limited set of Gestalt features' data and then the trained model is used to solve the conflicts between the partial gestalts. Using a machine learning method together with multiple Gestalt features[4] also makes the proposed method both novel and generalizable to other applications.

1.4 Hypothesis And Results

The proposed method tested if using multiple Gestalt features (similarity of intensity, common velocity and good shape) improves the detection performance of large

[3] The Gestalt law of connectedness: elements that are connected tend to be grouped together.

[4] Which are perceptually relevant, and this is important, since not all features are perceptually relevant [6].

oil sand lumps. The results obtained using different numbers of features were compared to each other. There was a tendency of better performance for a higher number of Gestalt features used in the experiments. More specifically, the proposed method had statistically better large lump detection performance with increasing number of Gestalt features in three out of four chosen performance measures.

The results confirm the prediction of Gestalt theory, which states that salient objects have a high probability to be detected using several partial gestalts [4]. Employing a machine learning method to model the relations between separate partial gestalts addresses the problem of Gestalt collaboration, which is still an open research topic.

1.5 Summary

The thesis addresses the general issue of how to detect global objects from local low-level information contained in the image. Unfortunately, the efforts of computer vision seem to be aimed at computing partial gestalts and the problem of feature combination (*i.e.,* a problem of partial gestalt collaboration and conflicts) is rarely addressed [7]. The work presented here uses three partial gestalts (*i.e.,* Gestalt features) to detect large lumps of oil sand. The problem of feature combination is addressed by using the machine learning inference engine. The novelty of the method is in its design and in its application. Design-wise novelty is that multiple perceptually significant features are used integrally with a decision tree-based model of feature interaction. The detection results obtained using the proposed method show experimentally that multiple Gestalt features improve detection performance, which was predicted by the Gestalt theory.

The next chapter will provide some background information on object recognition, segmentation. It will also discuss history and basic principles of Gestalt theory of psychology.

Chapter 2

Background Information

Object recognition is an important area in computer vision. Many computer vision techniques could be applied to perform object detection. Gestalt theory of psychology can enhance existing techniques with perceptual grouping laws for better object recognition.

2.1 Computer Vision Versus Image Processing

Computer vision and image processing are very important areas of computing science research. The main goal of computer vision could be defined as creating an artificial vision system that would mirror or exceed the performance of a human [44]. Computer vision contains both high-level and low-level processing of image data. High-level computer vision is typically knowledge-based and deals with object identification and detection. It uses many artificial intelligence methods. Low-level computer vision techniques are essentially image processing techniques. Image processing usually deals with extraction and conversion of digital image data. It does not involve recognition of objects or other similar ways of data interpretation.

2.2 Object Recognition Via Segmentation

Segmentation is one of the most complex problems in computer vision. It can be defined as dividing an image into parts so that there the parts correspond to objects or areas of the real world depicted in the image [44]. Object recognition applies already known information to segmented parts to select the objects of interest. Thus, object recognition can be viewed as segmentation combined with previous knowledge.

Sonka *et al.* [44] define and describe the following main categories of segmentation methods in computer vision:

1. Thresholding.
2. Edge-based segmentation.
3. Region-based segmentation.
4. Matching.
5. Mathematical morphology

Thresholding is probably the simplest segmentation method. The changes to the image data are performed based on the values of a single or multiple thresholds. Single thresholding involves one gray-level threshold value for the whole image (global thresholding). Multiple thresholds could be used for for different subimages (adaptive or local thresholding) for a range of intensities (band thresholding).

The proposed method works exceptionally well for well-defined homogeneous regions. It is also efficient and, therefore, is good for real-time processing. The result of thresholding is usually a binary image, but it also can be a gray-level output. For example, in semi-thresholding different gray-level thresholds can be set for different conditions.

Edge-based segmentation is another important way of segmenting images. A classical definition of an edge[1] is as follows: "an edge is the boundary between two regions with relatively distinct gray-level properties" [15]. The shortest edge can also be defined as a point in the image, where a significant intensity of texture change takes place. There are different methods to extract edge information. Edge

[1] In the thesis the term "edge" is used either to define a part of the object's contour visible by human eye or to define a connected component consisting of a set of 8-connected white or gray pixels. The term "contour" refers to the closed edge.

image thresholding uses edge-detectors to get an edge image, and then the edge image is thresholded. This method may have noise problems but that can be improved by combining spatially close edges together.

Region-based segmentation consists of region growing techniques and watershed segmentation. Region growing splits and merges subregions and needs some stopping criteria. Watershed method operates on region formation using local maxima (watersheds) and minima (catchment basins) values, and needs good initializations for these values. Region-based segmentation is sensitive to image noise. However, it requires the homogeneity of the regions to be quite high.

Other methods of edge segmentation include graph structures to search the images using dynamic programming techniques. A Hough transform could be used to identify certain edges: lines or circles, for example. Matching is usually used to locate objects by using already known data about the structure of the object: sub-images, patterns, or other descriptors and features. Mathematical morphology methods use nonlinear algebra and point sets to change image data. Morphological operations can successfully be used to segment images.

2.3 Gestalt Theory Of Psychology

Even given the most current advances in computer vision, artificial vision systems do not have the accuracy and depth of human perception. Humans are very successful at delineating various objects in complex scenes (*i.e.,* performing segmentation of the scenes). Therefore, it should be useful to look for object recognition ideas from the science that studies humans and how they see the world around them: psychology. It is important to know how humans detect objects in visual scenes, or, in other words, how they perform object recognition. The Gestalt theory of psychology is especially useful in this context. The theory discusses how human mind creates object perceptions from smaller elements. Gestaltists outlined a set of principles that govern element grouping. Several of these principles were used in the proposed method to show the importance of Gestalt ideas in computer vision.

2.3.1 History Of Gestalt Theory

Gestalt psychology emerged in Germany in early 1900s as a holistic theory alternative to the elementalistic bottom-up approach. The founders of Gestalt theory Max Wertheimer, Kurt Koffka, and Wolfgang Köhler were involved into research on apparent motion phenomenon: a person perceived motion when identical objects were sequentially displayed in different locations and each preceding object was deleted from the scene as soon as a new object appeared. In this experiment, the perception of motion was something more than just a sum of stimuli. Another example of apparent motion would be a movie: the viewer sees a sequence of static images on a film but perceives image contents as moving objects. The main inference was that the whole percept is not equal to the sum of its sensory parts [47].

Wertheimer using his experiments with several types of stimuli (lines, dots, etc) argued that humans perceive objects as unified wholes rather than elemental sensations [47]. Wertheimer published his "Productive Thinking" that was partially based on interviews with people known for their problem-solving abilities such as Albert Einstein who was Wertheimer's friend [49]. Wertheimer believed that problem solving should proceed from the whole problem down to its parts [47]. The assumption was that this approach would help to organize the details into meaningful form or Gestalt. This approach contradicted the behaviouristic idea of trial-and-error problem solving. Gestalt theory suggests learning the structure or relationship first instead of learning all the underlying details. For example, it is difficult to remember this sequence of numbers: 1 4 9 1 6 2 5 3 6 4 9 6 4 8 1 unless you notice that the numbers are just the squares of the numbers from 1 to 9.

Kurt Lewin, another famous Gestalist, applied Gestalt principles to the study of motivation, personality, and social principles by developing field theory, according to which a person interacts continuously within a field of psychological forces. The formula he used was $B = f(PE)$, where B was behaviour, P was a person, and E was environment. Lewin also studied group dynamics, within group communication and group decision process. His field theory influenced industrial psychology and personality theory [47]. Kurt Koffka studied child psychology from the Gestalt theory point of view. Wolfgang Köhler created the Gestalt learning principles and studied insight as a learning phenomenon performing research on animals [47].

The decline of Gestalt psychology can be explained by an unfortunate historical situation: when the Nazis came to power all the major Gestaltists departed from Germany for Scandinavia, Russia, North America or other countries leaving their students and well-established labs behind. Also, most of the founders of Gestalt theory had very few opportunities to supervise Ph.D. students. In addition to all that, their arrival to America coincided with the heyday of behaviourism and that, along with their criticizing of behaviourism [49] , did not help their careers in North America.

Henley and Thorne [47] state that by 1969 the Gestalt school of psychology was gone yielding a place to cognitive psychology that can be viewed as a combination of Gestalt theory and behaviourism. However, Gestalt theory still lives in social psychology and perceptual theory and it still continues to influence modern science, as it could be seen from its applications in computer vision field.

2.3.2 Basic Principles Of Gestalt Theory

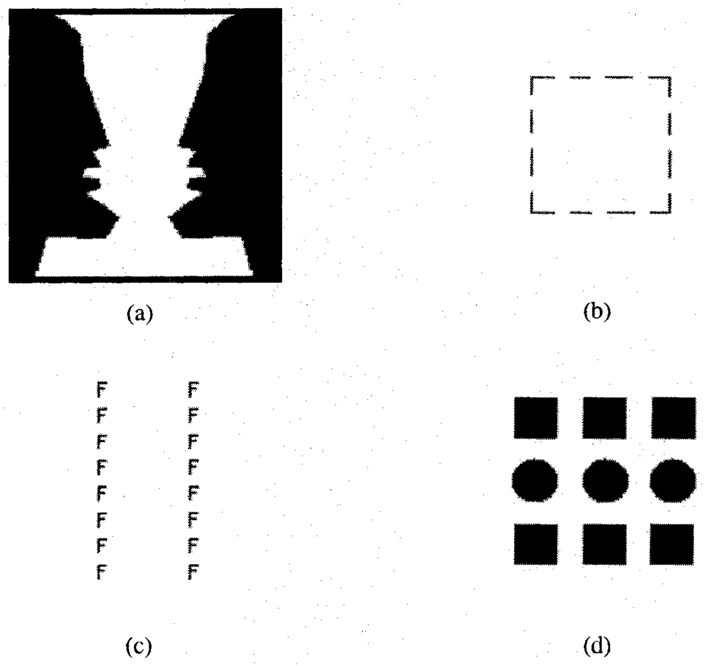
The concept of field lies at the heart of Gestalt theory [35]. This notion is adopted from Einstein's "field theory" and views phenomena (such as thinking, perception and mind in general) as arising from a network or field of forces as opposed to simplified cause-effect explanation of the nature of phenomena. The concept of field seems to determine the holistic approach of Gestalt theory.

The fundamental principle of Gestalt psychology is the law of Prägnanz. It asserts that we perceive observed phenomena as organized in the neatest, tightest, most meaningful way [35] or, in other words, as good Gestalt [47] in the given situation. The law of Prägnanz is a basis for all principles of perceptual grouping described further in the text.

Isomorphism (iso=identical, morphism=form) is one of principles. It was adopted from mathematical topology and assumes that conscious, phenomenological experience is isomorphic (or shares a common structure) with underlying physiological processes [35]. Wertheimer proposed that in the case of apparent movement (also known as phi-phenomenon) humans perceive motion because in this situation something isomorphic is happening at the physiological level that is similar to what would happen with the real movement. This implies that the nervous system does not have to possess an explicit mechanism of interlocking/successive elements for phenomena detection. This idea is consistent with recent advances in such area of machine learning as neural networks [47].

The Zeigarnik effect is also an illustration of the law of Prägnanz. According to this effect people tend to remember incomplete events better than complete ones (they strive for a good, complete Gestalt). Zeigarnik effect is used in Gestalt therapy that assumes incomplete tasks could be a reason for psychological dysfunction and the therapy itself is aimed at finishing this task (also known as completing the Gestalt in psychotherapy).

Harry Helson identified 114 laws of perception and thinking structure [17, 47]. Some of the principles discussed by [47, 35] are introduced below and examples can be observed in Figure 2.1.



(a) (b)

(c) (d)

Figure 2.1: An illustration of some of Gestalt principles of perceptual organizaton. Figure 2.1(a) illustrates the principle of figure-ground separation. Figure 2.1(b) shows the example of how the principle of closure works. Figure 2.1(c) is en example of the proximity principle. Figure 2.1(d) shows grouping by similarity of element shapes (redrawn from [14, 5, 36]).

Figure-ground relationship was borrowed from Danish psychologist Edgar Rubin and, according to it, the perceptual field is divided into figure and ground and the same object may either be figure or ground depending on if the observer is concentrated on this object or not [47]. In Figure 2.1(a) the observer can see two face profiles or a single vase depending which part of the image he/she concentrates on. The closure principle implies that there is a tendency to ignore minor breaks in a figure. Dashed line in Figure 2.1(b) are perceived as a square, even though its boundary consists of disjoint segments. The principle of proximity states that elements that are close to each other (in time or space) are seen as belonging together. In Figure 2.1(c) letters are seen as grouped vertically and not horizontally, since the vertical distance between letters is the smallest. The similarity principle says that elements, which are similar (in shape, color, texture, *etc.,*) are seen as belonging together. Figure 2.1(d) depicts an example of similarity, by the shape of constituent elements. The importance and application of Gestalt principles in object recognition will be discussed in the next chapter.

2.3.3 Gestalt Theory In The Proposed Method

Gestalt theory views the world in terms of objects (*i.e.,* Gestalts) that makes it interesting for object recognition. Gestalt laws of perceptual organization describe how object perceptions are created from smaller constituent elements. Gestalt laws could be important for successful object recognition, since they were perceptually significant (and, therefore, potentially generalizable to a number of different object recognition tasks) and corresponded to geometrical structure of the real world.

Several Gestalt laws have been chosen to be used in the proposed method. These laws were adopted as Gestalt features that were expressed by two or more parameters (see Table 2.1 for mapping between Gestalt laws, features and parameters and Figure 2.2 for illustration of chosen laws).

The law of similarity was chosen, because it was a very important law in Gestalt theory. The same reason was true for choosing the law of good shape. In addition, the benefit of using the goodness of shape was that it provided a high-level information about an object. The law of common motion was chosen, because it was

Gestalt Law	Gestalt Feature	Parameters
Similarity	Similarity of intensity	Mean intensity STD intensity
Common motion	Motion	Mean velocity Max velocity
Good shape (Prägnanz)	Shape	Solidity & Roughness (convexity) Eccentricity & Compactness (ellipticity) Extent (rectangularity/triangularity)

Table 2.1: The table shows mapping of Gestalt laws to Gestalt features and parameters used in the proposed method. More detailed explanation of parameters can be found in Section 4.2.4.

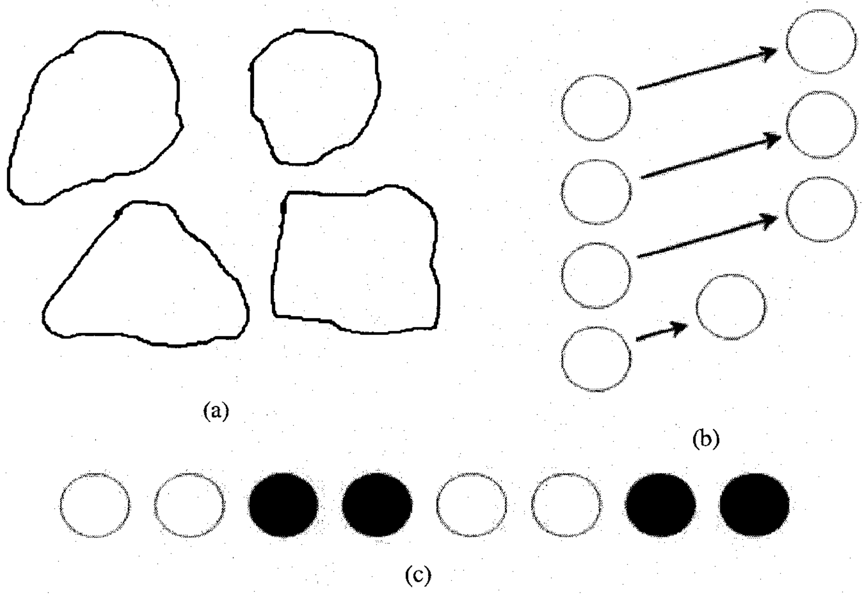


Figure 2.2: An illustration of Gestalt laws of perceptual organization that were used in the proposed method. Figure 2.2(a) illustrates the principle of good shape (the law of Prägnanz), where convex, elliptic, rectangular and triangular objects are preferred. Figure 2.2(b) is an example of the common motion law, where top three circles are grouped together, because they they have the same speed (indicated by the length of an arrow). Figure 2.2(c) shows grouping by similarity of element intensity (redrawn from [53]).

not usually used with other Gestalt laws, even though it seemed to have a lot of potential.

The Gestalt law of similarity was used as a Gestalt feature of intensity similarity. Intensity similarity implied that the brightness should be similar for large lump objects and different for non-large lump objects. The law of common motion was adopted as a Gestalt feature of motion and implied that the velocity of large lumps should be similar. The law of good shape (*i.e.,* the law of Prägnanz) implied that elements are organized in as good, balanced, simplest and efficient manner as possible. In the proposed method the goodness of shape (*i.e.,* a Gestalt feature of shape) implied convexity and ellipticity/rectangularity/triangularity of large oil sand lumps.

Another Gestalt law was used implicitly: the law of of closure was realized by creation of closed contours from disjoint edges (similar to the example in Figure 2.1(b)). Proximity was also used implicitely by limiting the active detection area with a region of interest.

2.4 Summary

The chapter gave some brief background information pertaining to the current work. Segmentation techniques for object recognition were discussed. The reader was introduced to the history and main concepts of the Gestalt theory of psychology. Gestalt laws used by the proposed method were also reviewed. The next chapter will discuss research literature that is relevant to the proposed study.

Chapter 3

Related Work

The usage of Gestalt ideas can be traced to early computer vision. However, the descriptive and abstract nature of Gestalt principles did not seem to facilitate their application in later research. Nevertheless, there were a few approaches that utilized at least some Gestalt features. In addition, some scientists conducted studies that quantitatively validated the importance of Gestalt principles. In this chapter some fundamental research related to Gestalt theory is reviewed that is followed by Gestalt theory applications in computer vision.

3.1 Selected Fundamental Research In Gestalt Psychology

In his work Harry Helson [17] discussed the fundamental ideas of Gestalt psychology. He also identified more than 100 laws that govern human perception.

He defined several results of object perception: configurations and totalities.[1] Configurations were denoted as segregated wholes that were governed by inner laws. A configuration was distinguishable from a totality which was a summative whole, whereas a configuration was an organic whole. Configurations were not seen by Helson as sums of their parts, and were not parts and relations between them, and did not have a strong dependency on the parts. Here Helson seemed to refer to the reconstructive ability of human perception, when object perception occurred even if sensory input is very limited. He mentioned that each configuration

[1] Helson seemed to refer to partial gestalts as "totalities" and to the whole Gestalts (*i.e.*, objects) as "configurations."

was based on a law/principle of organization or a principle of structure. If more than one configuration was possible from a complex of configuration, the more meaningful tended to be perceived. "Several configurations, if put together, may have one dominant configuration or may fuse", stated Helson, which seemed to imply a recursive nature of object perception.

Configurations might also be temporal. For example, a melody was described as a temporal configuration, as it needed time to be completed. A chord of melody was not a temporal configuration. Thus, dynamic processes were a basis of all configurations, and configurations were either an outcome of dynamic processes or are dynamic processes themselves. Configurations could be homogeneous, but Helson was not sure if they were configurations in that case, since there were no well-defined members or structure (single tone in music). Configurations could be complete or incomplete and combinations of elements could not be just called Gestalts, they needed the right conditions for that. Helson mentioned that goodness (meaningfulness) of perceived objects might change for the same configuration when different criteria were applied. Thus, Helson stated that using various partial gestalts changed the object's perception. Helson expressed an idea that configurations were not absolute and we could have levels of "gestaltness", though he did not mention how these levels could be identified.

Helson discussed many laws/principles governing human perception. Some of them are mentioned below. The law of inner necessity implied that a Gestalt usually changed due to inner forces. The law of Prägnanz suggested that a Gestalt "tends to become as good, precise, and impressive as possible." The law of simplicity meant that Gestalts tend to become as simple as possible; changes that occurred to Gestalts happened so that the least amount of energy was spent. The law of completeness was phrased as Gestalts tend to become complete in time.

Helson denoted the principle of figure and background separation, implying that every configuration must have a figure and a background. His definition was somewhat different from current understanding of the principle, which implied that a Gestalt can either be a figure or a background. Helson claimed that Gestalts have a tendency to resist changes that was expressed via the law of compensation and the law of interchangeability. The law of compensation suggested that "a change in one part of configuration can be made only at the cost of or change in another" and the law of interchangeability implied that some aspects of a Gestalt may have influence on another aspect. For example, brighter objects were judged to weigh less than darker objects. Objects that leave the ground with higher speed were decided to be lighter.

The law of symmetry was discussed by Helson as a tendency of Gestalts to symmetry, balance, and proportion. The tendency of Gestalts to leave after-effects was mapped to the principles of familiarity (familiar objects are perceived better than unfamiliar). The law of likeness (or similarity) stated that alike parts tend to form wholes. The law of nearness (or proximity) implied that parts that are closer together tend to be seen as wholes. The law of continuing curve stated that "if several possibilities are present by which a part may be continued in a whole, the simple, more regular will be chosen." The law of common fate argued that "any change in a part contrary to the general tendency of the whole will be resisted."

3.2 Selected Fundamental Research In Computer Vision

David Marr's book "Vision" [26] was at the very foundations of computer vision research [10]. Marr's approach was a natural computational approach to vision. Marr's computational approach studied the mind as an information processor. He applied natural constraints, such as rigidity, to object recognition theory. Marr realized that the functional analysis of the central nervous system performed in neurophysiology and psychophysics was missing something. He implied that it is more important to answer the question "why" instead of "what" or "how" something works.

Marr's emphasis seemed to be on a integrated approach to vision, despite some criticism for having too much emphasis on neurophysiology and binocular vision [10]. Marr stated that in order to understand the information processing device it should be analyzed at these three levels: computational theory, representation and algorithm, and hardware implementation. The strategy was to start from the computational level and work your way down. Marr proposed that the early visual processing (*i.e.,* determination of the components of the image) consists of two stages: raw primal sketch and full primal sketch. Raw primal sketch was viewed as a bottom-up process using such concepts as contour, texture, shading, occlusion. Full primal sketch was explained as a top-down process, which combined boundaries and regions into larger entities - surfaces. Marr used Gestalt grouping laws to explain how surfaces are created from their constituents.

In Marr's theory the early visual processing was followed by $2\frac{1}{2}D$ sketch. $2\frac{1}{2}D$ sketch was not yet a full 3D representation of the world. It gave information about the slant of the surfaces and about their relative depth. $2\frac{1}{2}D$ sketch had relative depth information but the scene in the mind was not yet perceived as 3D. It was based on the full primal sketch, retinal disparity (stereopsis), and structure from motion. Marr proposed using $2\frac{1}{2}D$ sketch to create the 3D model of the object by employing generalized cones (*i.e.,* using generalized conic shapes to model real objects) and matching them with the previous knowledge about 3D shapes.

Marr's work heavily influenced computer vision research. For example, Ullman [48], one of former Marr's students, also studied the structure from motion problem using Gestalt theory. He formulated the structure from motion theorem as follows: "given three distinct orthographic views of four non-coplanar points in a rigid configuration the structure and motion compatible with the three views are uniquely determined." The connection to Gestalt theory was in simplifying assumptions of object rigidity and orthographic views (that implies parallel perspective) that were viewed as the law of Prägnanz. The law of Prägnanz implied that human perception strives for as good an organization as the conditions allow. Thus, the law of Prägnanz may encompass many other Gestalt laws (for example, regularity, symmetry, simplicity, closure, good continuation, and good shape).

Ullman did not use motion cues (such as velocity and direction) to compute the structure from motion. He used geometrical cues (views and corresponding points) to create 3D structure. Ullman also talked about motion from structure, when the visual system fills in the perceptual gap between elements of 3D structure with motion. The condition for that was that the correspondences should be detected, which was done in accordance with the Gestalt principle of common motion. For example, one edge of a cube was highlighted. Later the edge was dimmed and a new edge having a common vertex with the old one was highlighted. The observer did not perceive two different edges, instead radial motion of a single edge around its vertex was observed.

Watson *et al.* [50] seemed to follow Marr's raw and full primal sketch stages while developing a model of velocity sensing in moving images by humans. Their model was applied to the Fourier frequency domain. Watson *et al.* started with the creation of a motion sensor model. They knew that human sensors do not assign an explicit value of speed at the threshold but that assignment happens if the speed is above the threshold. Using this knowledge the authors decided their sensor model would be based on two stages that corresponded to the at-threshold and above-threshold motion sensing. In the first stage the sensors (which were essentially a set of filters) that had the same scale and location but may have a different direction were grouped into the same velocity component (*i.e.*, velocity group). In the second stage the same direction responses of sensors were combined. Then the speed was calculated from the temporal and spatial frequencies, as the temporal frequency equals the dot product of the spatial frequency and the velocity. The drawback of their model was that it considered only translational motion.

3.3 A Conflict Between Computer Vision And Gestalt Theory Research

Infiltration of Gestalt theory into computer vision might have been slowed down by the very abstract nature of Gestalt ideas. Wertheimer [51] discussed the relationships between Gestalt-based perception and computer simulations. He suggested that computer models do not address the fundamental issues of insight and understanding and the emphasis should be made on productive (with deep understanding, for example, using prior knowledge) thinking when trying to solve any problem. He proposed a solution for successful computer systems implying that the following criteria should be satisfied: "(a) the representation corresponds to the actual structure of the problem (and this may be the crux of the issue), (b) the representation is well-integrated in the sense that all of its components are appropriately interconnected ... (c) the representation is well integrated with the problem solver's other knowledge". Max Wertheimer also suggested that similarity between objects is crucial for object recognition.

Wertheimer's article was later criticized by Simon [41], who stated that there were computer systems, which were more creative than Wertheimer suggested and could learn from given data, unlike Wertheimer stated. He also criticized the lack of operation definitions in Gestalt theory for such terms as intuition, insight, understanding, and good Gestalt.

The ideas discussed by Gestaltists indeed seem to be quite abstract. However, further research was successfully able to extend the ideas to different application domains and even quantitatively measure the importance of some of the Gestalt principles.

3.4 Non-Probabilistic Methods

The methods described below did not require *a priori* knowledge of image content to perform successful detection/segmentation of salient structures. The saliency was provided by perceptual and geometrical relevance of Gestalt analysis. Combination of Gestalt principles was typically done by applying iterative methods to mathematical formulas.

3.4.1 Salient Region Extraction Using The Centre Of Gravity Partial Gestalt

Ma and Zhang [25] used the Gestalt principle of the centre of gravity to identify most general salient regions (*i.e.,* attended views) in arbitrary images. The salient or attended views were defined as image rectangles having the most contrast information. Regions of interest were extracted from images using contrast features. Gestaltists said "visual forms may possess one or several centers of gravity about which the form is organized" ([40], p.113). Therefore, Ma and Zhang used this idea to find the salient view (*i.e.,* one "most interesting" area that is a centre of gravity of the saliency map, determined via the method of moments) that is later subdivided into smaller areas. An interesting observation was made that humans were more perceptive to changes in smooth areas than in textured areas.

However, using contrast to measure the centre of gravity does not explain the issue of *masking*, when object perception changes depending on where attention is concentrated. For example, in Figure 2.1(a) we can choose to see either a vase or a face (depending where we concentrate), even though contrast information does not change.

3.4.2 Multiscale Detection Of Perceptually Relevant Objects

Tabb and Ahuja [45] proposed a multiscale image segmentation method, where the final segmented image contained only the most perceptually relevant structures. The authors defined perceptually relevant structures as ones that appeared at multiple scales. The method used homogeneity within an increased range of neighbourhoods as a main criterion. Homogeneity and space scales were the main parameters of the proposed segmentation model. Homogeneity scale was defined as the difference (sum of differences) between a pixel and its neighbourhood. Space scale was identified as the neighbourhood of a pixel. Homogeneity and space scales selection determined the zone of attraction: inward force vectors. As space and homogeneity scales increased the attraction zones appeared and disappeared. Edges were detected in attraction zones as places, where vectors along the boundaries diverged from each other. Structures (*i.e.,* segmented objects) were identified as closed boundaries.

Tabb and Ahuja claimed that their method performed Gestalt analysis, since it did not make any assumptions about the objects' geometry and used similarity to perform segmentation. Edge detection was performed using neighbourhood and homogeneity (similarity) notions and the region was detected as the area within the closed boundary. Therefore, region detection seemed to be a result of edge detection, not a parallel process.

An important advantage of the method introduced by Huart and Bertolino [18] was that it used multiple scales for object segmentation. First, a region-growing technique was applied to image pixels. The image was presegmented into homogeneous regions using a colour distance threshold. Region-growing was applied iteratively to the local pyramid graph structure until no more homogeneous regions (homogeneity was based on a colour distance threshold) were identified. Second, the homogeneous regions from step one were further grouped applying Gestalt principles of proximity, similarity, closure, continuity and symmetry.

3.4.3 Region-Growing Using Combination Of Common Motion And Intensity Similarity

The method described by Lee *et al.* [22] was region-based and used a region-growing algorithm to perform segmentation. Region growing was based on the watershed algorithm. Motion and intensity information was used to create watershed seeds and to merge the subregions. The details of their approach were as follows. The segmentation started with the second frame of the image sequence, since the first frame was used for motion field estimation. First, the method created seeds for region-growing. The intensity criterion was used to extract initial seeds via a homogeneity measure based on the manual threshold. Intensity seeds were then refined (using motion information also based on a manual threshold) into more seeds. The seeds, which were obtained using intensity and motion, were further used in watershed region-growing. Region growing also combined intensity and motion information to create a distance measure used to create final segmentation. Temporal tracking was used after the first couple frames to improve segmentation results. The method assumed that there was "no important change in the scene contents", since temporal tracking would fail if drastic changes in motion were present between two consecutive frames.

Even though the method proposed by Lee *et al.* [22] did not explicitly mention Gestalt features it could be viewed that the authors used Gestalt features of similarity by intensity and common motion. A strong side of the method was that Lee *et al.* combined motion and intensity information to segment objects. A drawback of the method seemed to be the necessity to set parameters manually, and this usually makes a method very application-dependent.

3.5 *A Posteriori* Approaches

The methods described below rely on *a posteriori* knowledge of image content. Partial gestalts were typically combined using previous knowledge along with probabilistic or machine learning techniques.

3.5.1 Contour Modelling

Zhu [52] introduced an approach that creates a learning-based model of an object's contour. The model is defined and trained on the Markov random field[2] (an MRF). Here MRF was defined as a structure (*i.e.,* field) containing probabilities of the shape, where each probability corresponds to a point on object's contour and encapsulates several Gestalt laws: co-linearity, co-circularity, parallelism, symmetry, and proximity. Co-linearity and co-circularity implied grouping of elements that formed a line or a circular arc. Parallelism/symmetry implied grouping of elements that were parallel/symmetrical with respect to each other. The Gestalt laws were defined using notions of curvature (contour information) and distance (region information) on artificially extracted primitives (linelets). The model learning was performed using MRF created from the statistical data of already known contours.

The main problem with the method seemed to be that it was computationally expensive. Zhu found an experimental evidence that the principle of co-linearity was more important than the principle of co-circularity. The learned models of object's contour could be used to generate the random shapes that are matched to the initial shape or some other shape. The results could also be used for object recognition and image segmentation, as in the work by Litvin and Karl showed [24].

Litvin and Karl [24] used a probabilistic shape modelling method developed by Zhu [52] to perform segmentation of noisy data. A shape model was trained on two sets of non-noisy images: one set contained sharp-cornered shapes and the other one contained smooth shapes. Curvature was used to define MRF probabilities,

[2] An important property (Markov property) of a Markov random field is that each probability in the field depends only on the probabilities of its neighbours.

although Litvin and Karl did not mention Gestalt laws specifically. Gaussian noise was added to the binary image (one of the images from each training set), which was used for segmentation using a known shape model. Segmentation was performed by "maximizing the posterior density for the shape given the data" on two data sets. The results were good since segmentation with prior shape information produced smoother boundaries and good corners versus segmentation that did not use a shape model. The main problem of the method was high computational price. Also, there seemed to be a drawback that both training and test images were artificial, which did not allow to make any claims about suitability of Gestalt features in natural scene segmentation.

3.5.2 Extraction Of Salient Edge-Based Structures

Maßmann *et al.* [28] also used a manually labelled training set and Gestalt principles to select most salient contours. The initial input consisted of edges. Edges were then approximated by arcs and lines (*i.e.,* image primitives). Arcs and lines within areas of certain size and shape (*i.e.,* areas of perceptual attentiveness) were used to form groups of primitives. Areas of perceptual attentiveness were essentially masks that restricted the number of primitives' combinations. The number of groupings was further reduced by applying orientation and distance thresholds to arcs and lines of each group. Areas of perceptual attentiveness and thresholds were determined from the hand labelled training set. Application of the areas of perceptual attentiveness and orientation and distance thresholds seemed to be the way of authors' implementation of Gestalt laws of proximity and good continuation (curvilinearity, collinearity). The groupings using laws of symmetry/parallelism and closure were created using a proximity graph, where each node corresponded to a curvilinear or collinear grouping. The most salient groups of lines and arcs were further selected using MRF energy function minimization, which resulted in a final segmentation containing most salient structures.

Thus, Maßmann *et al.* [28] addressed the problem of partial gestalt conflict by creating a fixed hierarchy based on a principle's complexity. The lowest level was occupied by proximity, collinearity and curvilinearity. The middle level contained symmetry and parallelism. The most complex top level consisted of the laws of closure (creation of a final object's contour). The conflicts between Gestalt principles were solved using MRF. The main problem with the method seemed to be the need to manually determine thresholds (from training data) for Gestalt laws.

Sarkar [38] also applied machine learning and Gestalt laws to extract salient edges from digital images. The following Gestalt principles were used: proximity, similarity, continuity, co-circularity, and parallelism. The image primitives were edge segments represented as arcs and lines. Each pair of primitives was defined by a weighted combination of Gestalt laws applied to the pair. The value of the weighted combination of each edge pair was extracted using Bayesian inference on a set of training images, which consisted of original images and ground truth segmentations. Weighted combinations for each pair were put into a graph. The larger groups were formed from the graph partitioning using its eigenvectors. The input into the Bayesian networks was computed for every pair of lines and arcs and consisted of: min/max distance between two given primitives, min distance between the endpoints of primitives, overlap, slope difference, and photometric attributes. There were different networks for lines, arcs and corresponding Gestalt laws. For example, for lines their parallelism, continuity, T-junction, L-junction were characterized by min/max distance between two given primitives, min distance between the endpoints of primitives, overlap, and slope difference. For arcs parallelism and co-circularity were defined by min/max distance, min endpoints distance, and overlap. Proximity was predicted by min endpoints distance. Region similarity was predicted by photometric attributes.

It seemed that Sarkar [38] used manually set weights for each Gestalt principle to identify its strength. Thus, the problem of the conflict between different Gestalt principles was addressed by explicitly assigning Gestalt feature importance, which did not seem to be justified. Sarkar determined that the good continuation principle in tested images (variable, from different domains) did not play an important role and had "low ability to segregate objects from each other and from the background". He also found that proximity and similarity were important Gestalt features.

3.5.3 3D Spatio-Temporal Edges In Object Detection

Korimilli and Sarkar [21] and Sarkar *et al.* [39] used Gestalt principles of proximity, continuity and parallelism to segment long (containing more than twenty images) sequences of digital images. They represented each sequence of images as a $3D$ spatio-temporal volume. A $3D$ version of a classical Canny edge detector (see Section 4.2.1 for more information on a Canny edge detector) was applied to the volume. Resulting edges were approximated and grouped into temporal regions using the Hough transform. Temporal regions were further grouped (now using Gestalt features and Bayesian inference) to form so called temporal envelopes. Each temporal envelope corresponded to a single moving object. The strong points of the method were as follows: (1) the authors used Gestalt principles for both spatial and temporal grouping (for example, the partial gestalt of parallelism in time implied the law of common motion: if two or more lines were parallel in time that mean they were moving in a similar fashion and could be grouped together to form a single object); (2) the method used Bayesian networks to combine partial gestalts. The method seemed to work quite well with noisy data, multiple moving objects, occlusions and changing illumination.

3.5.4 Region-Growing Segmentation Based On Gestalt Features

Ren and Malik [33] proposed a linear regression classification technique that used Gestalt laws to segment natural images. They used following Gestalt principles: similarity of texture and intensity, good continuation, closure and proximity. Good continuation was defined by the tangents' difference in a point between two superpixels/regions. Similarity was defined for texture, intensity and contour energy within and outside each region. Closed contours were determined via computing each pixel's orientation energy.

The method was as follows. First, similar to the paper by Huart and Bertolino[18], the image was oversegmented into a predetermined number of superpixels (small homogeneous regions). Second, Gestalt cues were computed for each region. Third, the regions are grouped iteratively, taking a random region each time. The grouping was done using linear regression learning that solved an optimization problem for a function that maximized the sum of Gestalt features' values. The training group consisted of a number of human-segmented images (from [27]) depicting natural scenes. Ren and Malik found that presence of real-world edges in the region-grouping hypotheses was the most important cue for image segmentation. Good continuation and similarity were found to be very important Gestalt features too.

3.5.5 Application-Tuned Usage Of Gestalt Features

Nattkemper *et al.* [31] applied selected Gestalt features to detect more or less convex/circular objects (tissue cells) in images obtained using fluorescence microscope. Nattkemper *et al.* proposed to segment images via binding chosen features (image coordinates and intensity gradients) to salient groups. Saliency of the groups was measured using the following Gestalt principles of perceptual organization: co-circularity[3] and convexity.[4] Binding was performed training a neural network on image coordinates and intensity gradients. Intensity gradient of a pixel could be defined as a direction of maximum growth (*i.e.,* a direction in which some neighbourhood of a pixel changes the most). Experimenters obtained the image gradient using simple edge detection. The Gestalt principle of convexity was described by collinear or co-oriented gradients. The measure of co-circularity was provided by collinear and reverse-oriented gradients. Thus, the method proposed by Nattkemper *et al.* produced figure-background separation via grouping pixels into salient groups using Gestalt features of co-circularity and convexity. The method appeared to be resistant to image noise. However, using neural networks might be a drawback, since they may require more parameters to be set.

Galkin *et al.* [13] applied the Gestalt principles of proximity, smoothness and cocircularity within a neural network framework to preclassify a database of radar images of Earth's magnetosphere (*i.e.,* plasmagrams). The objects of interest in plasmagrams were line-like or smoothly curved structures. Therefore, grouping of plasmagram primitives to form final objects called traces was done using continuity,

[3] In Gestalt theory co-circularity implied element grouping if they formed an arc.

[4] Convex objects were considered to be good Gestalts (a specific case of the law of good shape).

smoothness and proximity partial gestalts. Continuity and smoothness were measured based on orientation of primitives. The general structure of the method was also consistent with David Marr's vision paradigm [26].

3.6 Other Applications Of Gestalt Theory

The usage of Gestalt ideas was not limited by vision problems. For example Chang *et al.* [5] used a wide set of Gestalt laws for visual screen design: symmetry, continuation, closure, figure-ground, focal point, isomorphic correspondence, Prägnanz, proximity, similarity, simplicity, and harmony. They also introduced an additional law of focal point, which implies that the points that are different from the surroundings are points of interest from which the viewer starts visual processing and follows further in his/her observation (this law could also be called a principle of dissimilarity). Chang *et al.* also reintroduced the law of familiarity as the law of isomorphic correspondence (it implies the importance of previous knowledge in interpretation of the visual scene).

Reybrouck [34] discussed using Gestalt ideas in music analysis. Reybrouck stated that perception of separate pieces of music was governed by Gestalt grouping laws. He emphasized the importance of temporal Gestalts, where the layers of music were viewed in terms of Gestalt figure-ground separation.

3.7 Analysis Of Gestalt Laws

Martin *et al.* [27] were able to show quantitatively that the usage of Gestalt laws in object recognition is objectively justified. They showed quantitatively that Gestalt laws of proximity, similarity and convexity were useful for object segmentation and recognition (which they considered the main problems in computer vision). They used a set of ground truth segmentations of natural scenes (50 images, 150 segmentations by 10 people, each image was segmented 1-5 times by different people, time to segment was less or equal to 5 minutes, number of segments in each image was 2-20) to validate numerically (using the probability theory) that the aforementioned Gestalt laws are applicable to computer vision domain. More specifically, they have shown that the probability of points belonging to the same objects increases when: (1) distance between them decreases (the law of proximity) and (2) intensity between them decreases (similarity). They showed that intensity by itself is not enough to segment/represent a generic natural scene object (a maximum of 60% of pixels pairs were identified correctly as belonging to the same object based on their intensity). They have also shown that there were many more segmented objects that had a high convexity (which they computed by the mathematical formula of solidity) compared with objects with lower convexity. Thus, their results confirmed that the chosen Gestalt laws work for natural images.[5]

Ben-Av *et al.* [2] studied how humans perform visual grouping and found that using the law of proximity is most important, when stimuli are presented for a very short time. However, similarity seemed to be a more important principle than proximity, when more time was allowed for visual processing (which is the case in digital image processing). Max Wertheimer [51] suggested that the principle of similarity is crucial for object recognition. He also predicted the importance of prior knowledge for a successful computer model, which might explain a higher number of "a posteriori" methods that use Gestalt features compared to non-probabilistic methods.

Martin *et al.* [27] stated that only the good continuation law has been shown to be useful scientifically via usage of probability distributions. Their results were consistent with observations of Ren and Malik [33]. However, Sarkar [38] found that the good continuation principle had a "low ability to segregate objects from each other and from the background." Cao [4] also stated that in natural images there was a smaller number of good continuations, since natural images were very irregular. Thus, the good continuation law (unlike other Gestalt principles) received conflicting assessments from different researchers.

Martin *et al.* [27] supported the idea that there is only a limited amount of research on Gestalt laws. Their opinion was consistent with views of Desolneux *et al.* [9, 10, 6, 7] who thought there should be more attention to Gestalt principles and

[5] The importance of convexity and intensity was later used in the thesis algorithm for parameter selection.

their usage in computer vision research.

3.8 Renaissance Of Gestalt Theory In Computer Vision

Research conducted by Desolneux *et al.* [9, 10, 6, 7] seemed to have a potential to give a powerful impulse to further development of Gestalt theory in computer vision. They reformulated Gestalt theory within computer vision not only in terms of grouping principles, but also raised such important issues, as partial gestalt collaboration, conflicts, masking,[6] their recursive nature and the necessity of using multiple partial gestalts.

3.8.1 *A Contrario* Approach To Gestalt Theory

Desolneux *et al.* [8] proposed a systematic approach "aimed at adapting Gestalt theory to Computer Vision" ([4]), p.1). Their partial gestalt computation was based on the Helmholtz principle, which implied that the relevant object can be found by comparing their structures to random noise. Essentially, they used random noise as a prior for their model (hence, *a contrario* approach). The method checked if having a certain feature (for example colour or orientation) for a certain group of objects was a coincidence or not, which seemed to be in agreement with the Gestalt principle of meaningfulness. Desolneux *et al.*proposed minimal description length to address partial gestalt collaboration and conflicts.

They showed experimentally that human detection of objects also operates in a way similar to Helmholtz principle. The importance of previous knowledge was incorporated into their model as the law of past experience. One of method's advantages was that it had only a single threshold that had to be set up manually.

Cao [4] successfully applied the method of partial gestalt creation proposed by Desolneux *et al.*to detection of perceptually relevant smooth boundaries. He created a specialized edge detector that detected good continuations and corners/terminations using Helmholtz principle. Thus, he defined a model of an irregular random curve

[6] A phenomenon that occurs when one partial gestalt completely dominates object's perception (also might be known as the Gestalt principle of emergence).

using the Helmholtz principle and employed the model to detect curves that did not follow the model.

Cao mentioned that the Gestalt principle of good continuation could be used for motion analysis, since trajectory of a moving object was a smooth curve. However, the idea seemed to have already been implemented by Korimilli and Sarkar [21] and Sarkar *et al.* [39] (see Section 3.5.3).

Another contribution of Cao's work was that he essentially created an alternative edge detector - a detector that did not explicitly use high contrast to detect boundaries. It detected good continuations in level lines.[7] Therefore, the extracted boundaries were not edges by image processing definition (which were defined by Cao as high contrast lines). They were a different type of boundaries, since they were non-contrast edges which were also smooth. Cao noted that the coincidence of his good continuation edges with perceptual edges was very good. Cao [4] also stated that results were not sensitive to smoothing, which implied that the method would work well for multiscale edge extraction.

3.8.2 Importance Of Gestalt Theory In Contemporary Computer Vision

Desolneux *et al.* mentioned that "computer vision used very little and almost nothing of the Gestalt theory results" ([8], p.1), even though "grouping is the main process in our visual perception" ([8], p.2) and "not all geometric structures are perceptually relevant; a small list of relevant ones is given in Gestalt theory" ([6], p.3).

The main problem with computer vision seemed to be that its efforts in using Gestalt theory were mainly aimed at computing different partial gestalts [8], leaving out the fact that collaboration of partial gestalts and their conflicts were "seldom addressed" ([7], p.3). Sometimes it might have seemed that a partial gestalt (or a feature) was enough, but the final results were always incomplete since "in natural world images partial gestalts often collaborate" ([7], p.4). Thus, a single Gestalt

[7] They are contrast invariant curves - pixel subsamples extracted from the level sets of a grayscale image, where level sets are essentially graylevels of a quantized (every 3,5, or 10 levels are compressed together) image.

feature could provide a fix to a limited problem but probably not to a general problem. They also mentioned that good detection could not be a result of partial gestalt summation. It needed to be a synthesis, since some gestalts could be stronger and some could be weaker [10]: "only a global synthesis of all partial gestalts can give the correct result" ([7], p.21) .

Ideally, there should be a program that would first compute all known gestalts and, second, combine them according to their hierarchy [4]). However, it is an ongoing research. The problem of partial gestalt collaboration is theoretically and computationally challenging and interactions of features are nonlocal according to Cao [4]).

3.8.3 Importance of Multiple Partial Gestalts

According to Gestalt theory multiple partial gestalts are very important for successful object detection: "objects that are conspicuous[8] are very likely to be detected by several partial detectors (as predicted by Gestalt Theory), and a single detector does not give a definitive answer." ([4], p.12).

Desolneux *et al.* also emphasized the importance of using multiple Gestalt features: "most salient objects or groups come to sight by several grouping laws ... The outcome of a partial gestalt detector is valid only when all other partial gestalts have been tested and the eventual conflicts dealt with" ([10], p.20).

3.9 Summary

There were many perceptually and geometrically relevant Gestalt principles identified, which played a significant role in early computer vision. Many computer vision methods used Gestalt principles for object recognition and segmentation. Not many computer vision researchers seemed to apply full Gestalt theory knowledge. Many studies were limited to using partial gestalts for element grouping only, without referring to such important issues of Gestalt theory as partial gestalt collaboration, conflicts, masking and the importance of using multiple Gestalt features.

[8] Obvious to the eye or mind [19].

There seemed to be a limited amount of research that studied and compared multiple partial gestalts. The latter was unexpected, since using as many as possible partial gestalts was an important point in Gestalt theory [4].

Most of the reviewed methods used either spatial or temporal Gestalt grouping. There were both region-based and edge-based methods, which used Gestalt features. However, there seemed to be more edge-based methods, which could be usually explained by higher computational cost of region-related algorithms. Also, according to Ren and Malik [33], edges might be the most important cue in natural images.

Many reviewed methods used prior knowledge (which is an important Gestalt principle according to Max Wertheimer [51]) to fuse different partial gestalts. Their drawback might be that, without re-learning and resetting of the parameters, those methods are not very generalizable to other applications. On the other hand, non-probabilistic methods did not seem to be very strong in object segmentation/detection. Their output could be considered as good postprocessed input for recognition algorithms.

Analysis of Gestalt features showed that similarity was a very important partial gestalt, which was followed by the proximity principle [27, 33, 38, 2, 51]. Evaluation of the good continuation principle received conflicting assessment from different authors [27, 33, 38, 4]. The usage of the motion within Gestalt framework did not seem to be common. Korimilli *et al.* [21] and Sarkar [39] stated that only few scientists used Gestalt principles with motion data.

Shape-related Gestalt features also were deemed to be quite important. For example, Desolneux *et al.* [7] mentioned the law of similarity of shape as one of the main perceptual grouping principles. The work of Martin *et al.* [27] showed that convexity (*i.e.,* solidity) is a very important shape feature too.

Some methods used application-specific Gestalt features. For example, Nattkemper *et al.* [31] used features that were tuned to detect elliptic objects (closed contours), whereas Galkin *et al.* [13] used features that favoured continuity and smoothness (disconnected objects).

The next chapter will describe the structure of the proposed object detection method, which is based on using multiple Gestalt features.

Chapter 4

Gestalt Features Model

This chapter describes the proposed model of large lump detection. The proposed method uses Gestalt features of intensity similarity, common motion and goodness of shape to detect large lumps of oil sand.

4.1 Assumptions

It was assumed that large lumps were moving and the camera was stationary. It was also supposed that there were no significant problems with input data: no steam was present in images and lighting conditions did not vary a lot.[1] It was expected that large lumps are not covered by oil sand material (*i.e.,* there were no occlusions). It was assumed that large lumps have wider (*i.e.,* thicker) edges than smaller pieces of oil sand ore. It was decided that large lump edges should be strong enough to appear in two consecutive frames.

4.2 Proposed Method

The outline of the proposed method is shown in Figure 4.1 (a detailed diagram is depicted in Figure 4.2). First, an edge detector is used to find edges in two consecutive frames. Second, edges between frames are matched and only those edges are selected that appear in both frames. Edge velocity is also computed during that step. Third, regions are formed based on edges from the previous step. Chosen

[1] Some images that did not have significant problems were still difficult to segment even for human experts.

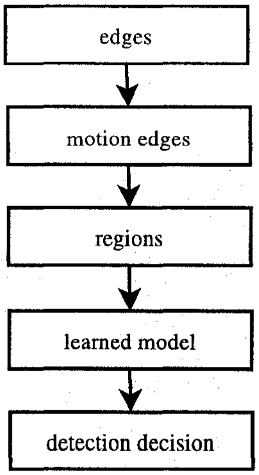


Figure 4.1: The outline of the proposed method. Each box corresponds to a single module in the main application.

parameters, being quantitative measures of Gestalt features, are computed for each region.[2] Then a learned model is applied to each region's parameters to determine if it could be a large lump. If at least one region is labeled as having a large lump, then the entire frame (the first of two) is marked to have a large lump.

4.2.1 Edge Extraction

Edges were used as basic input elements (*i.e.*, primitives). Wider edges were considered more important. Canny edge detector (discussed in the next subsection) was chosen to extract edges being both a reliable and optimal edge detector that was also capable to extract edges at different scales. Canny was also used because it is quite resistant to edge noise: the method does not discard weaker edges if they are connected to stronger ones. To the human eye edge images seemed to contain enough information for identification of large lumps (see Figure 4.3 for an example).

[2] The parameters of the model are in accordance with Gestalt principles. Motion parameters are used in accordance with the Gestalt principle of common motion (*i.e.*, motion parameters express the Gestalt feature of motion). Intensity parameters are used in accordance with the Gestalt principle of similarity by intensity (*i.e.*, intensity parameters express the Gestalt feature of intensity). Shape parameters are used in accordance with the Gestalt principle of good shape (*i.e.*, shape parameters express the Gestalt feature of shape).

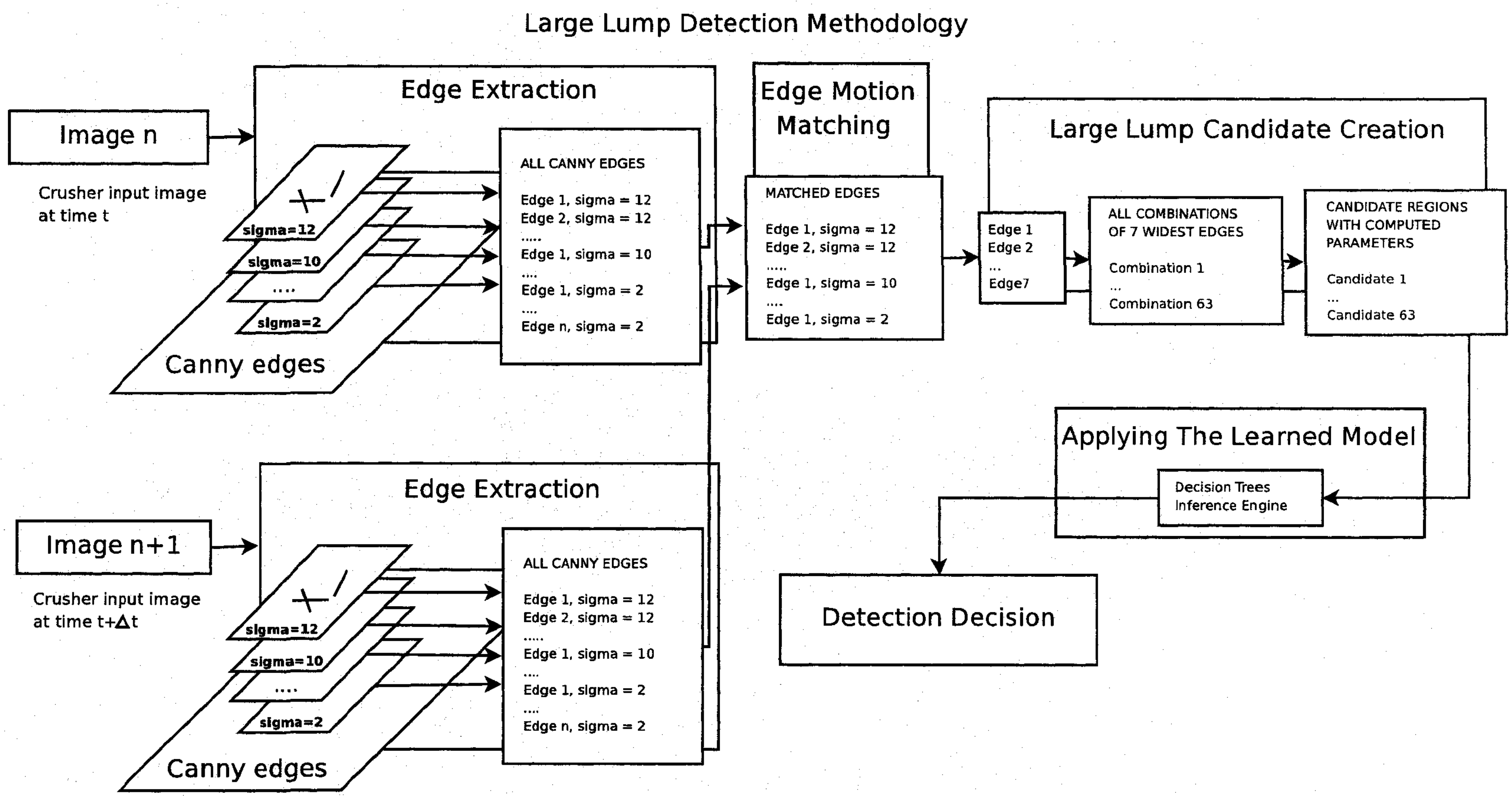


Figure 4.2: A detailed description of the proposed method. Initial input consists of edges extracted from two consecutive grayscale images. The edges of the first image are matched to the edges of the second image to form motion edges (*i.e.*, edges that appear in two consecutive frames). Then, up to seven top edges (see Section 4.2.3 for information on why certain number of edges was selected) are then used to create up to sixty-three candidate contours (each contour delineates a candidate region). Gestalt features' parameters are then computed for each candidate region (a detailed explanation of parameters can be found in Section 4.2.4). Finally, a decision tree is applied to Gestalt parameter data to filter large lumps out.

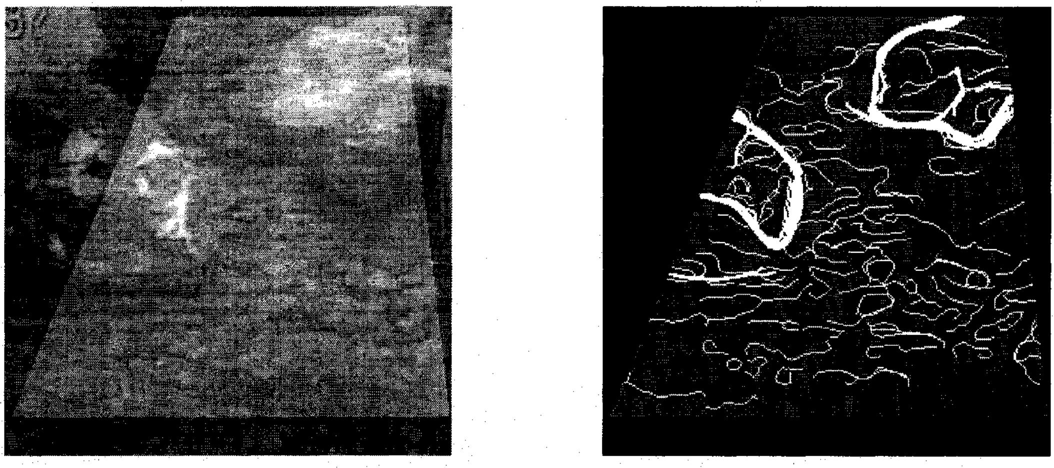


Figure 4.3: The original image (top) and edges (bottom) extracted from it at six different Gaussian smoothing levels (*i.e.*, scales).

The edge extraction algorithm is shown in Figure 4.4.

Description Of Canny Edge Detector

Canny edge detection is one of the most robust and widely used edges detection methods. It has a very good precision and efficiency. Canny edge detector is an optimal edge detection method, which compares favourably to other edge detection techniques [1, 16]. In the proposed method a Matlab [29] implementation of a Canny edge detector was used. The Canny method detects the edges by using the local maxima of the intensity gradient of the input image.

First, the magnitude of the gradient[3] (expressed as $|\nabla I|$) is computed at each pixel of the input image I. $|\nabla I|$ is defined as follows:

$$|\nabla I| = \left\| \begin{bmatrix} \frac{\partial I}{\partial x} \\ \frac{\partial I}{\partial y} \end{bmatrix} \right\| = \sqrt{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2} \tag{4.1}$$

where $\frac{\partial I}{\partial x}$ and $\frac{\partial I}{\partial y}$ are partial derivatives of I along X-axis and Y-axis, respectively. They are computed as follows:

$$\frac{\partial I}{\partial x} = I(x, y) * \nabla G_x \tag{4.2}$$

$$\frac{\partial I}{\partial y} = I(x, y) * \nabla G_y \tag{4.3}$$

[3] The image gradient ∇I at a specific image location is usually defined as the rate of intensity change at that location.

```
Algorithm $ExtractEdges(I_n)$
Input: $I_n = 640x480$ array of 8-bit pixels; a grayscale image $I_n$, which is a video
    sequence frame number $n$
Output: $L_{edges_n}$: a variable size list that consists of edges extracted from frame
    number $n$, which are sorted in decreasing order from widest to narrowest.
(* Demonstrates how edge extraction module works *)
$\sigma \leftarrow 12$
while $\sigma \geq 2$
    do
        compute a Canny edge image $I_{n_c}$ from $I_n$ using $\sigma$ as a Gaussian smoothing parameter
        extract all 8-connected edges $L_{edges_\sigma}$ from $I_{n_c}$
        add $L_{edges_\sigma}$ to the list of all edges $L_{edges_n}$
        $\sigma \leftarrow \sigma - 2$
return $L_{edges_n}$
```

Figure 4.4: An algorithm describing the edge extraction module of the proposed method. σ is the scale parameter for Gaussian smoothing, which controls the width of extracted edges. Higher value of σ implies more smoothing, and, therefore, more wider edges to be detected. The range of σ values from 12 to 2 and with the step -2 were determined experimentally.

where ∇G_x and ∇G_y are the directional derivatives of 2D Gaussian function along X-axis and Y-axis, respectively. ∇G_x is computed as follows:

$$\nabla G_x = -\frac{c}{\pi\sigma^2} \exp^{-\frac{c^2+r^2}{2\sigma^2}} \tag{4.4}$$

where c and r are columns and rows numbers of a square matrix, where each row is equal to the vector t.[4] ∇G_y is computed as a transpose of ∇G_x. The input image I is also smoothed by derivatives of Gaussian. The rate of smoothing is controlled by the σ value.

Gradient's magnitude values of the image are further normalized by their maximum value:

$$|\nabla I| = \frac{|\nabla I|}{\max |\nabla I|} \tag{4.5}$$

Second, the Matlab implementation applies non-maxima suppression to thin the

[4] t is a vector of values from -1 to 1 (3 elements) up to -30 to 30 (61 elements). t is computed based on the width of the Gaussian filter. The width of the filter is computed using σ, which is preset by the user. In the proposed method following σ values are used: 2, 4, 6, 8, 10, 12. These values were determined experimentally using the training sample mentioned above.

gradient edges. Each positive ∇I is checked against its two neighbours along the direction of ∇I. If the value of ∇I is smaller or equal then it is set 0.

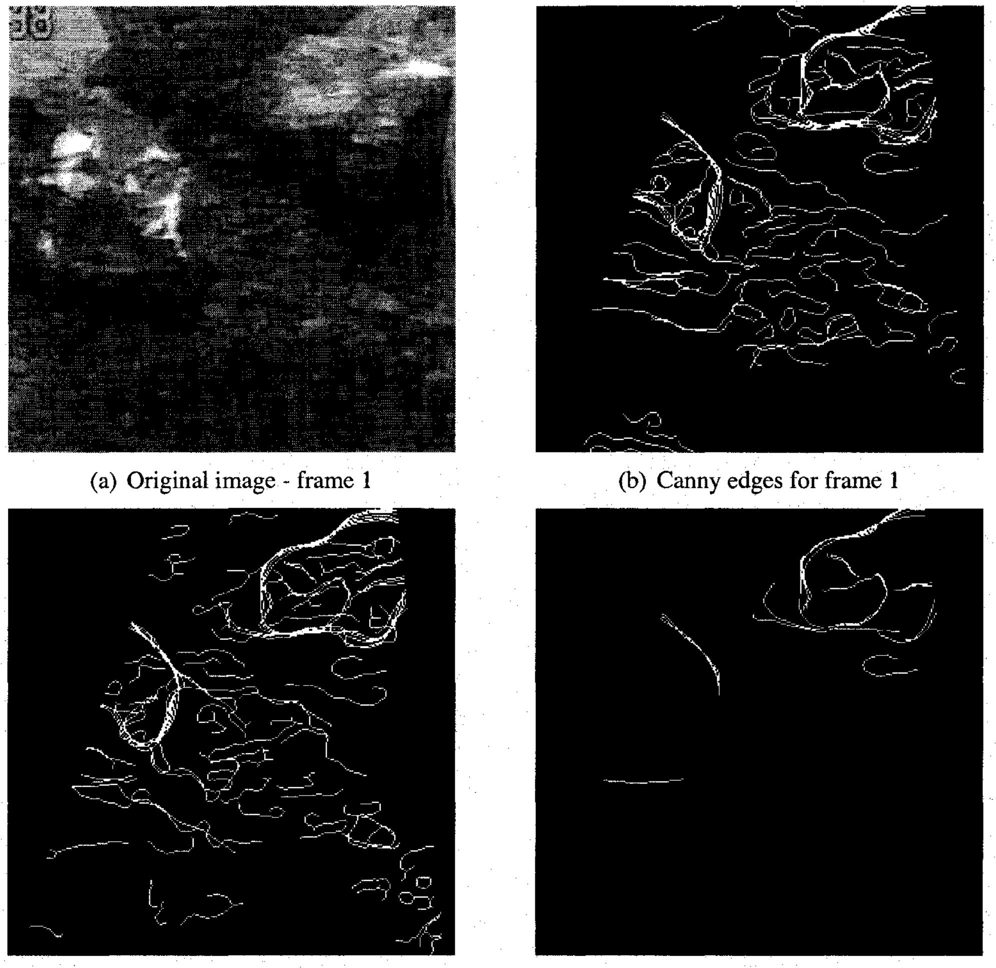
Third, a lower T_1 and a higher T_2 thresholds are computed. Matlab creates an intensity histogram of the gradient. T_2 is set to be the value corresponding to a bin with the highest number of members that is then $0.7\times$(size of I along X-axis) $\times$ (size of I along Y-axis). T1 is set to $0.4 \times T2$. Computed thresholds are used to create two binary images: image T_1 includes weaker edges and image T_2 includes stronger edges. In the final image edges from T_1 (weaker edges) are included only if they are connected to the edges from T_2 (stronger edges). Thus, more accurate edges are obtained by including both stronger edges and weaker edges in the final edge output. However, stronger edges must be connected to weaker edges for the latter to be included.

4.2.2 Motion Matching

Edge matching between two consecutive frames was used to select the more important edges (ones that appear in two consecutive frames) and to compute the velocity of edges that was later used for a motion Gestalt feature. A combination of edge characteristics (*i.e.,* a signature) was used to perform the matching. An example result of motion matching is shown in Figure 4.5. Motion matching seems to work especially well with images that do not contain large lumps. It tends to eliminate false or weaker edges (see Figure 4.6). The algorithm of motion matching is shown in Figure 4.7.

Matching Using Edge Signature

A unique representation for each edge was needed, so that the edge could be correctly matched to another edge in the next frame. It was decided to test the idea that a certain number of edge descriptors would create a unique 1-D signature of that edge from Gonzalez *et al.* [15]. Following descriptors were used: coordinates of the centre of gravity of the edge, its orientation, solidity, eccentricity, extent, major axis length, mean intensity within a convex hull of the edge, and a standard deviation of



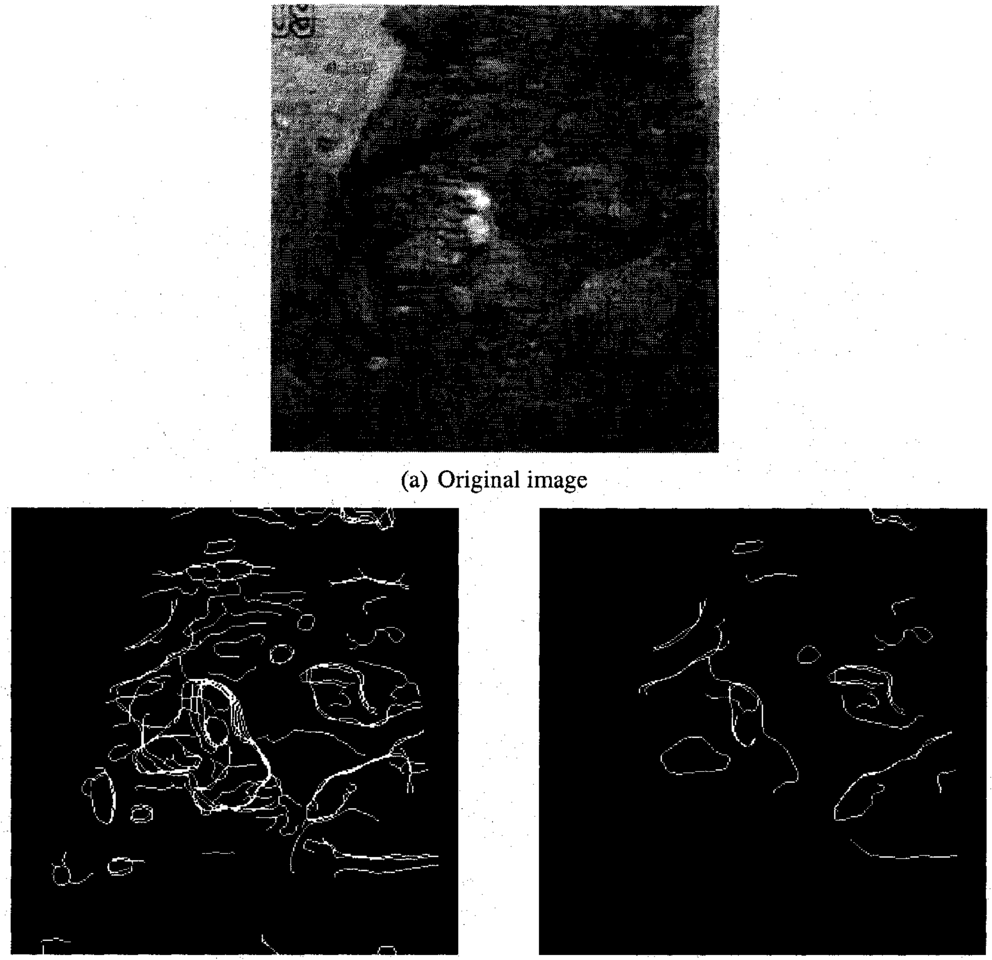
(a) Original image - frame 1

(b) Canny edges for frame 1

(c) Canny edges for frame 2

(d) Motion matched edges for frame 1

Figure 4.5: Motion matching for large lump images. Motion matching (Figure 4.5(d)) leaves only the strongest edges - ones that appear in both frames (the first frame is displayed in Figure 4.5(b) and the second one is in Figure 4.5(c)). Original frame 2 is not shown here. Both frames were shot using the same camera. 1 second elapsed between frames. Note: Some edges (that appear matchable to the human eye) might not be matched, since they may have spurious connections (not easily visible in images shown here) that prevent proposed 1D signature matching.



(a) Original image

(b) Canny edges

(c) Motion matched edges

Figure 4.6: Motion matching for non-large lump images. Motion matching (Figure 4.6(c)) eliminates some false edges that appear after Canny edge detection (Figure 4.6(b)).

```
Algorithm MatchEdges($L_{edges_n}$, $L_{edges_{n+1}}$)
Input: $L_{edges_n}$ and $L_{edges_{n+1}}$ returned by the algorithm in Figure 4.4.
Output: A list of edges $L_{moving\_edges_n}$ that contains edges that appear in both frames
    $n$ and $n + 1$; a list of corresponding edge velocities $V_n$.
(* Demonstrates how the edge matching module works *)
compute normalized edge descriptor vectors $d_{n+1}$ for all edges in $L_{edges_{n+1}}$
for $e$ is an edge in $L_{edges_n}$
    do
        compute its normalized edge descriptor vector $d_e$
        compute SSD errors $errors_{SSD}$ between $d_e$ and each descriptor vector
        contained in $d_{n+1}$
        $minSSDError \leftarrow \min(errors_{SSD})$
        $b \leftarrow$ an edge corresponding to $minSSDError$
        if $minSSDError \leq 15\%$
            compute Euclidian distance $v_e$ between the centres of gravity of
            $e$ and $b$
            normalize $v_e$
            add $e$ to $L_{moving\_edges_n}$
            add $v_e$ to $V_n$
return $L_{moving\_edges_n}$, $V_n$
```

Figure 4.7: An algorithm describing the extraction of moving edges from two consequtive frames. The following descriptors were used for each edge descriptor vector: coordinates of the centre of gravity of the edge, its orientation, solidity, eccentricity, extent, major axis length, mean intensity within a convex hull of the edge, and a standard deviation of intensity within a convex hull of the edge.

intensity within a convex hull of the edge.[5] The descriptors were computed using standard Matlab [29] functions.

The choice of descriptors was motivated by uniqueness of edge's shape, location and size. The values of the descriptors were normalized by their maximum values in the traning sample. Matching was performed by comparing edge descriptors and choosing ones with the smallest sum of squared differences that did not exceed some threshold (determined manually by trial-and-error).

Motion Matching Evaluation

The performance of the proposed matching technique was analyzed using one image with 40 matchable edges (Figure 4.5(b)). The analysis resulted in 8 true positive matches, 26 true negatives, 2 false positives and 4 false negatives. Thus, the accuracy (computed using Formula 5.1 from Section 5.3) of matching was 0.89, which was quite good, but not enough to make a strong claim. Therefore, matching accuracy was qualitatively assessed by having a human expert to look at several randomly selected images. Matching also appeared to be good in the latter case.

Velocity Computation

For each pair of matched edges the velocity of the first edge was approximated as a distance between their centres of gravity. The centre of gravity (or centroid) was computed using a standard Matlab [29] function. The centre of gravity is well-described by Sonka *et al.* [44]. For binary objects it is a sum of either x or y pixels coordinates divided by the number of pixels [44].

There was also an issue of velocity and scale normalization. Objects that were closer to the camera appeared to be wider and have a higher velocity than objects situated farther away. The issue was addressed by assigning a normalization coefficient to each horizontal line of the image. Computed edge velocity and the

[5] Solidity, eccentricity, extent, mean intensity and a standard deviation of intensity were computed for the edge in the same way as outlined in Section 4.2.4. Edge's orientation and major axis were computed as, respectively, orientation and major axis of an ellipse fitted over the edge. X/Y-coordinates of edge's centre of gravity were defined as the sum of edge pixels' X/Y coordinates divided by the number of pixels in the edge. The actual computations were performed using corresponding Matlab [29] functionality.

maximum dimension of a large lump[6] were multiplied by the coefficient's value to achieve normalization.

The value of the coefficient NC for each horizontal line k was computed by dividing the width of the largest conveyor belt opening (closest to the camera) $Largest\_Width$ by the width of the belt at the location of that horizontal line $Width_k$:$NC_k = \frac{Largest\_Width}{Width_k}$.

4.2.3 Candidate Region Creation

To create a region from edges, every edge endpoint was connected to every other edge endpoint (except ones that belong to the same edge) together. The chosen method of endpoints connection created a shape that preserved all main concavities and convexities of constituent edges. Edge images seemed to contain enough information for identification of large lumps. The algorithm of candidate region creation is shown in Figure 4.8.

Motivation For Chosen Method

Initially, candidate shapes were created via fitting a convex hull over their edges. However, it was later noticed that the configuration of candidate edges was not always convex, and fitting a convex hull sometimes would give an incorrect impression that the resulting shape is convex (see Figure 4.9). Therefore, it was decided that a new method of shape creation from an edge combination (*i.e.,* configuration) was needed, a method that would not cover up the true shape of the configuration of edges (see Figure 4.10).

Edge Connection

The proposed method of edge connection connected all endpoints of all edges in a configuration, except endpoints from the same edges. Every edge configuration consisted of 1 to N edges. N was chosen to be 3, because there were usually no more than 3 edges detected for a single large lump outline according to observations.

[6] The maximum dimension of a large lump was used in the final detection decision module to decide how large the lump really is.

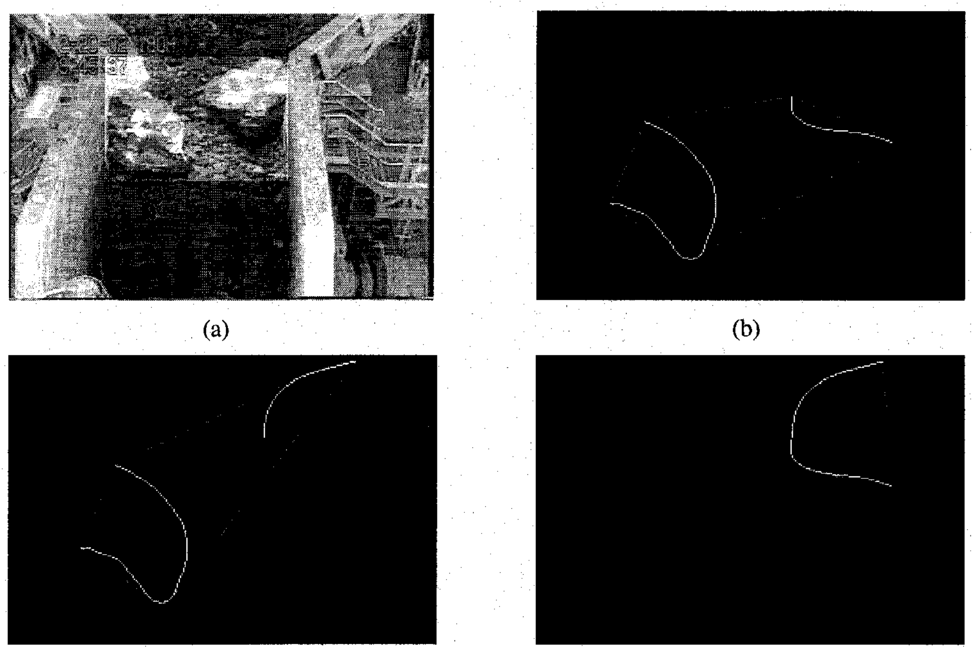
Algorithm *CreateCandidateRegions*$(L_{moving\_edges_n}, V_n)$

Input: A list of motion matched edges $L_{moving\_edges_n}$ for frame n; a list of corresponding edge velocities V_n.

Output: A list C_n of candidate regions' contours for frame n; a list I_n of corresponding intensity parameters; a list S_n of corresponding shape parameters; a list M_n of corresponding motion parameters.

```
(* Demonstrates how the candidate creation module works *)
for $i \leftarrow 1$ to 7
    do
        for $j \leftarrow 1$ to 7
            do
                for $k \leftarrow 1$ to 7
                    do
                        connect all endpoints of $L_{moving\_edges_n}(\text{UNIQUE}(i, j, k))$
                        fill in the interior of the resulting region
                        extract the contour $C_{n_{i,j,k}}$ of the filled region
                        compute intensity parameters from $C_{n_{i,j,k}}$ and add them to $I_n$
                        compute shape parameters from $C_{n_{i,j,k}}$ and add them to $S_n$
                        compute motion parameters from $V_{n_{i,j,k}}$ and add them to $M_n$
                        add $C_{i,j,k}$ to $C_n$
return $C_n, I_n, S_n, M_n$
```

Figure 4.8: An algorithm describing the creation of candidate contours for large lump detection. The algorithm searches through all unique combinations of 7 widest edges, where each combination can have up to 3 edges (the choice of numbers 7 and 3 is discussed in a paragraph before Section 4.2.5).



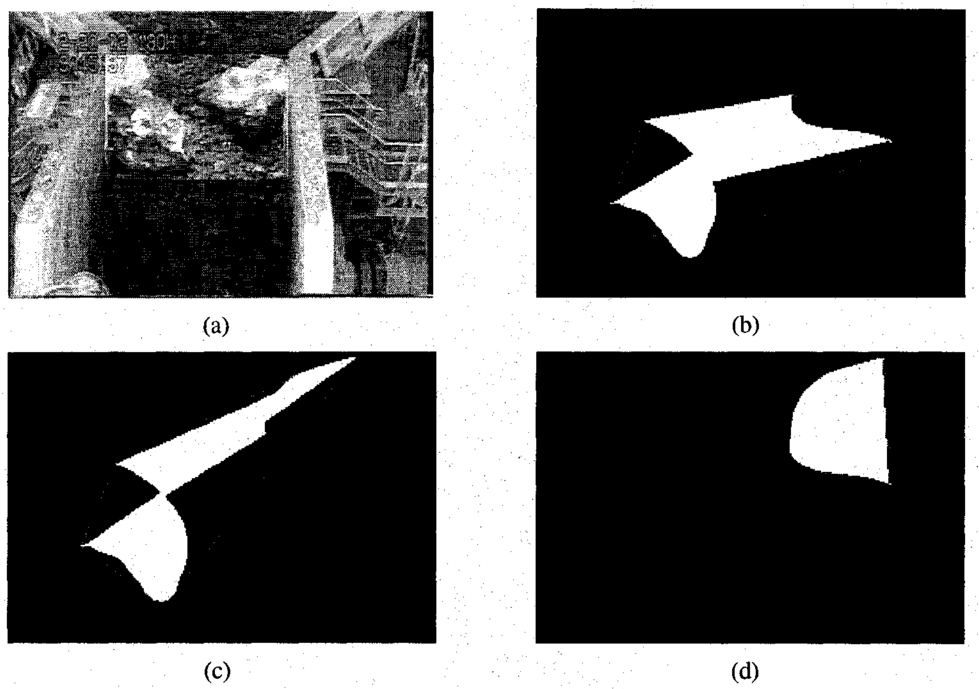
(a)

(b)

(c)

(d)

Figure 4.9: The marked area in 4.9(a) corresponds to the edge images in 4.9(b), 4.9(c), and 4.9(d). The darker line denotes a convex hull fitted over the candidate region's edges. The input edges are depicted in white. It can be observed that using a convex hull (instead of the proposed edge connection method) falsifies the true shape of edges.



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

Figure 4.10: The marked area in 4.10(a) corresponds to the edge images in 4.10(b), 4.10(c), and 4.10(d). Solid white region corresponds to the candidate that was created via connecting its edges' endpoints (a proposed method). The darker contour around white regions corresponds to the candidate's contour that was created via fitting a convex hull over its edges. Notice how the proposed edge connection method performs well in 4.10(b) and 4.10(c).

From observations 3 edges were usually enough to identify a large large lump in the given set.

Only widest M edges participated in region creation. "Top widest edges" implied most important edges, in the sense that they were the widest among those that were detected. When using the motion feature, the top edges importance was reinforced by an additional requirement that each of them had to appear in at least two consecutive frames. M was chosen to be 7, because this number seemed to provide sufficient edge information and triple combinations of 7 edges also yielded a computationally reasonable number of 63[7] possible candidates for each image. For example the algorithm would have to go through 41 triplets if using top 6 edges, 92 triplets if using top 8 edges.

4.2.4 Selection Of Features And Parameters

Gestalt features and parameters (via which features are numerically expressed) were selected using perceptual psychology ideas, relevant literature review and experimental observations. Following Gestalt features were selected: similarity of intensity, common motion and goodness of shape. Each feature was represented by two or more parameters. The intensity similarity Gestalt feature contained mean and standard deviation of intensity inside the large lump candidate. The motion feature consisted of mean and maximum velocity of edges. The shape feature consisted of roughness, solidity, eccentricity, compactness and extent of the candidate's contour.

Preliminary Analysis Of Large Lump Events

To understand the nature of the problem large lump events were first visually analyzed using following characterization: size, shape, presence of rotation, smoothness of texture, flatness, smoothness of contours, and amount of edge information of large lumps present in the frame. It was found that observed large lumps had more or less elliptic shape, but could also be triangular, rectangular or oblong. Therefore, ellipticity was not enough by itself. A number of lumps were rotating while

[7] The number of combinations for 7 edges with up to 3 members is computed as follows: $\binom{7}{3} + \binom{7}{2} + \binom{7}{1} = \frac{7!}{(7-3)!3!} + \frac{7!}{(7-2)!2!} + \frac{7!}{(7-1)!1!} = 35 + 21 + 7 = 63$.

progressing through the apron feeder. Most lumps had uneven texture, smooth and well-defined edges. The results of observation can be viewed in Figure 4.11. The data from the analysis was later used in shape parameters selection.

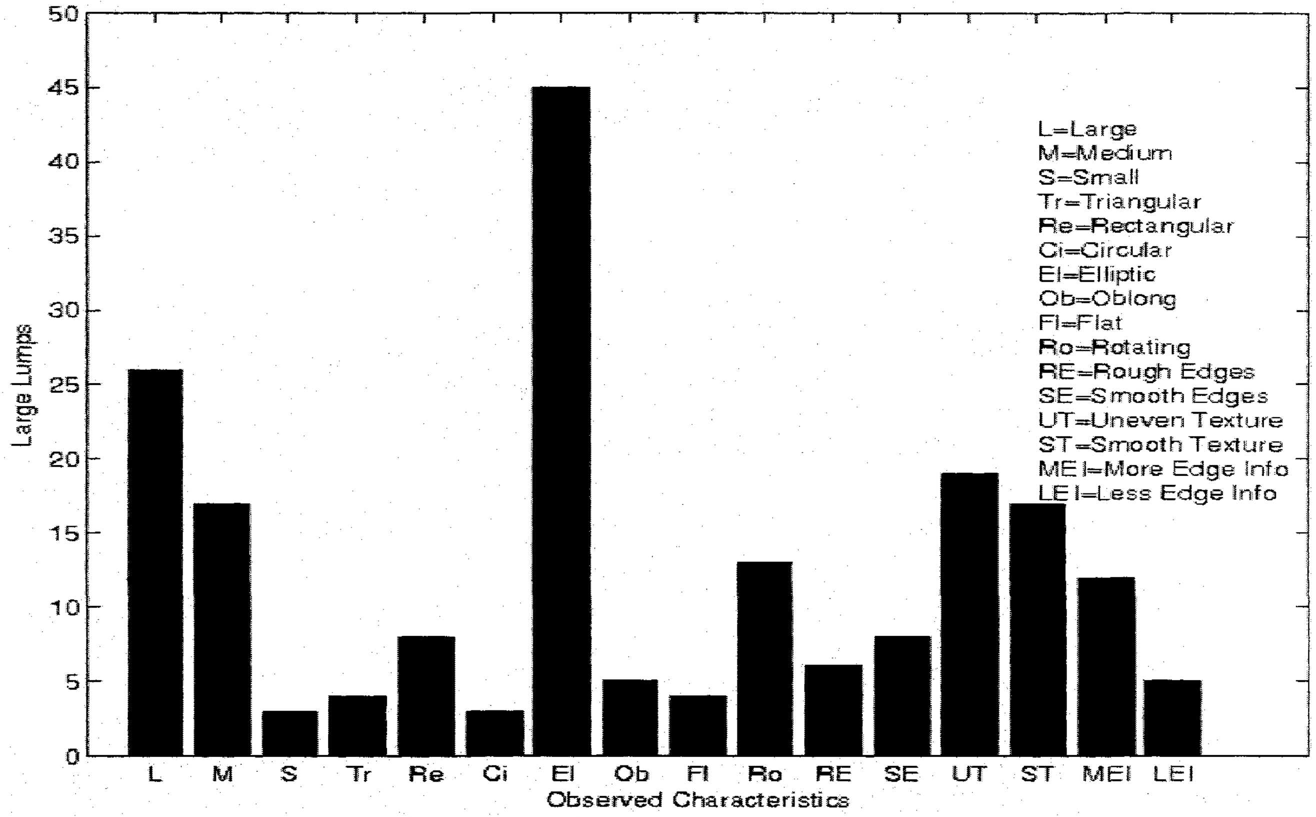


Figure 4.11: Visual analysis of large lumps. 46 large lump events were analyzed by characteristics of their large lumps: maximum number of lumps in the event, size of lumps (large - more than 2.5m in maximum diameter, medium - more than 1.5m, small - more than 1m), their shape (triangular, rectangular, circular, elliptic, oblong, flat), if lumps rotated while moving along the apron feeder, smoothness of their texture and contours, and by how well lumps' edges were defined.

Parameter Explanation And Computation

Mean intensity (*i.e.*, $Intensity_{mean}$) was computed as an arithmetic mean of all pixel intensities contained inside the candidate region. *Standard deviation of intensity* (*i.e.*, $Intensity_{STD}$) was set to be a standard deviation of intensity of all pixels inside the candidate region.

Mean velocity (*i.e.*, $Motion_{mean}$) was an arithmetic mean of all constituent edges' velocities. *Maximum velocity* (*i.e.*, $Motion_{max}$) was set to be the maximum of all constituent edges' velocities.

Roughness (*i.e.*, $Shape_{rough}$) was computed as the ratio of candidate's perimeter over its convex hull's perimeter. *Solidity* (*i.e.*, $Shape_{sol}$) was estimated as the ratio of candidate's area over its convex hull's area. Thus, roughness and solidity were measures of candidate's convexity: the lower the roughness and the higher the solidity were, the higher the convexity of the candidate large lump was. Growing solidity implies increasing convexity, while growing roughness implies decreasing convexity.

Eccentricity (*i.e.*, $Shape_{ecc}$)was computed as the ratio of the distance between foci of an ellipse fitted over the candidate and the length of candidate's maximum diameter. Foci are such points on the major axis of an ellipse that the sum of distances from these points to any one point on the ellipse is constant [11]. Eccentricity may be computed as

$$\sqrt{1 - \frac{minorAxisLength^2}{majorAxisLength^2}} \tag{4.6}$$

Compactness (*i.e.*, $Shape_{comp}$) was computed as follows:

$$\frac{4 \times \Pi \times Area}{Perimeter^2} \tag{4.7}$$

Eccentricity and compactness both measure how circular the candidate is. Eccentricity of 0 and compactness of 1 show that the candidate is circular. Maximum value for eccentricity is 1 and maximum value for compactness is not limited.

Extent (*i.e.*, $Shape_{ext}$) was computed as the ratio of the candidate's area and its bounding box's area. Note that the bounding box was defined as a minimal bounding box is the direction of the x-axis. It was expected that extent will help to detect rectangular or triangular shapes.

An important property of chosen parameters is that they are invariant to candidate's rotation and scaling. A more detailed explanation of shape parameters could be found in work of Kindratenko [20].

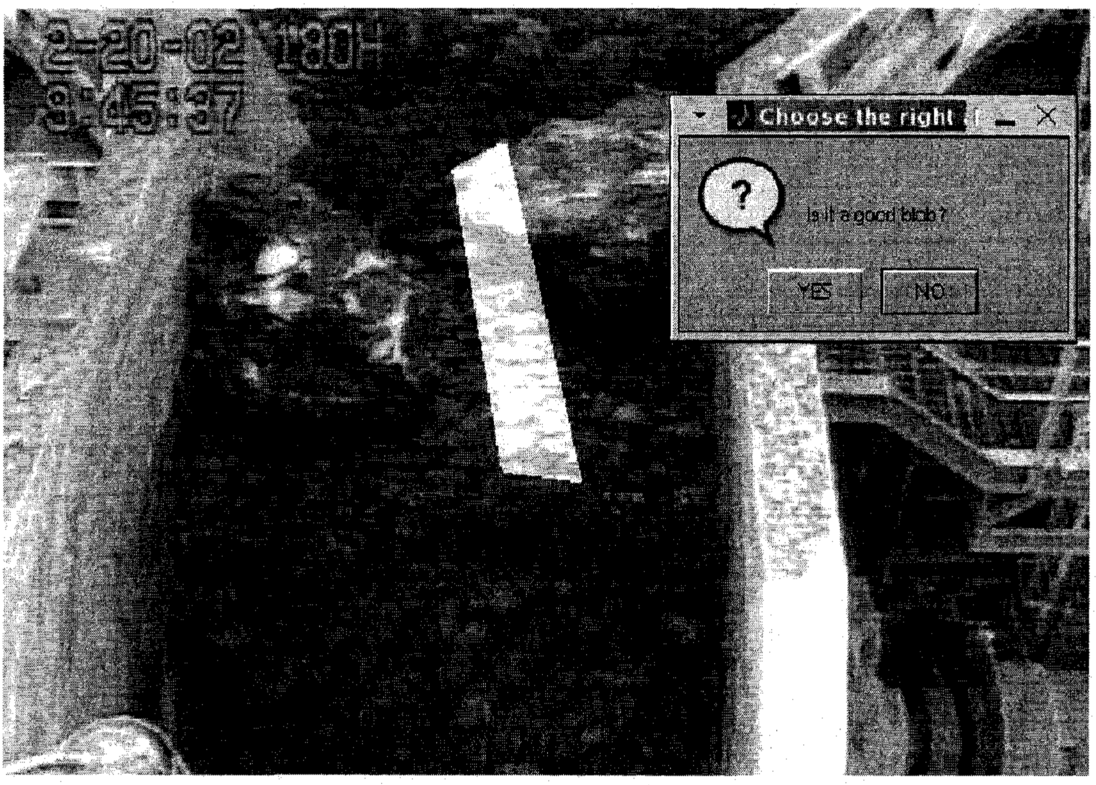
Parameter Statistics Of Training Sample

The parameter statistics for large lump region and non-large lump regions was analyzed using the training sample. The goal was to determine if two groups had significant parameter differences. The parameter statistics (see Table 4.1) confirmed the fact that chosen parameters are important features for large lump detection. Computed statistics could also be used to choose manual parameter thresholds.

It was decided to characterize the parameters using a 40-image sample. The goal was to determine if chosen parameters were sufficient to differentiate between large lumps and smaller (or nonexistent) lumps. The idea was to determine if there are significant differences in parameters for regions that corresponded to large lumps ("good" candidates) and for regions that did not correspond to large lumps ("bad" candidates). A GUI was created that allowed the experimenter to manually label each lump candidate as good or bad (see Figure 4.12). Only those candidates were considered, whose major axis length was at least half of the chute width. A good lump candidate was assumed to have at least 90% of its area inside the actual large lump, had to cover at least 50% of the large lump and had to have reasonable edges (*i.e.*, the edges had to be located on or close to the real edges of the lump).

Using the GUI a set of lump candidates was created that consisted of 67 "good" large lump candidates and 913 "bad" candidates. These candidates were further used to compute the parameter statistics and, later, to train decision tree models.

It was assumed that the samples of good and bad large lumps were representative of their true population. The computed parameter statistics can be observed in Table 4.1. Means correspond to the average value for a corresponding parameter for either "good" or "bad" class. The standard deviation was computed for each parameter. 95% confidence intervals were derived for the means. 95% confidence interval implied that in 95% of "good" and "bad" samples the true population mean would be inside the respective interval. Therefore, if the confidence intervals for



Is it a good blob ?

Figure 4.12: A screenshot of a GUI used to select "good" and "bad" regions, which are later used to create parameter statistics. Each lump candidate was labeled as "good" or "bad" given the candidate's region (lighter area in the image) and original grayscale image.

Parameters	**"Bad" Large Lumps**			**"Good" Large Lumps**		
	mean	**STD**	**95% CI**	**mean**	**STD**	**95% CI**
$Intensity_{mean}$	122.4	20.84	[121; 123.7]	149.7	31.41	[142.2; 157.3]
$Intensity_{STD}$	9.35	0.8341	[9.295; 9.404]	9.468	0.686	[9.303; 9.632]
$Motion_{mean}$	13.58	7.73	[13.08; 14.08]	12.47	6.606	[10.89; 14.05]
$Motion_{max}$	20.48	15.55	[19.47; 21.49]	15.7	8.406	[13.69; 17.72]
$Shape_{rough}$	1.02	0.09114	[1.014; 1.026]	1.007	0.0148	[1.004; 1.011]
$Shape_{comp}$	1.947	1.198	[1.869; 2.025]	1.602	0.6207	[1.453; 1.75]
$Shape_{ecc}$	0.8679	0.1227	[0.8599;0.8758]	0.8428	0.1562	[0.8054;0.8802]
$Shape_{sol}$	0.8391	0.1585	[0.8288;0.8494]	0.9054	0.06264	[0.8904;0.9204]
$Shape_{ext}$	0.4617	0.1481	[0.4521;0.4713]	0.5711	0.1272	[0.5407;0.6016]

Table 4.1: Parameter statistics computed from manually selected blobs.

some classes for the same parameter did not intersect, it could be stated (with 95% confidence) that those classes belong to different distributions.

Thus, information in Table 4.1 was used to show that the "good" and "bad" regions are really different. It could be stated with 95% confidence that "good" and "bad" samples belong to different distributions with respect to the following parameters: mean intensity, maximum velocity, roughness, compactness, solidity and extent.

Raw Data Plots

Raw data (see Figures 4.13 and 4.14) showed that a single parameter is not enough to separate large lumps from non-large lumps.

It was decided to plot raw parameter data using normalized histograms. Normalization of data was performed as follows. The number of elements in its largest bin was determined for each sample. Then a normalization coefficient was computed by dividing the larger number of elements by the smaller one. Then all frequencies (*i.e.,* number of elements) in bins of the sample with smaller number of elements were multiplied by the normalization coefficient. As a result, the maxima of both histograms were set to be equal and histograms became comparable.

Class separation was confirmed by data histograms for extent and mean intensity (see Figures 4.13 and 4.14). There seemed to be no visible separation for other parameters, which showed that most probably single parameters would not be sufficient for accurate detection of large lumps. Using combinations of parameters could yield better results. Parameter interactions could be visualized in two- or three-dimensional space. However, it seemed to be impossible to visualize all chosen parameters, since such visualization would imply nine-dimensional space.

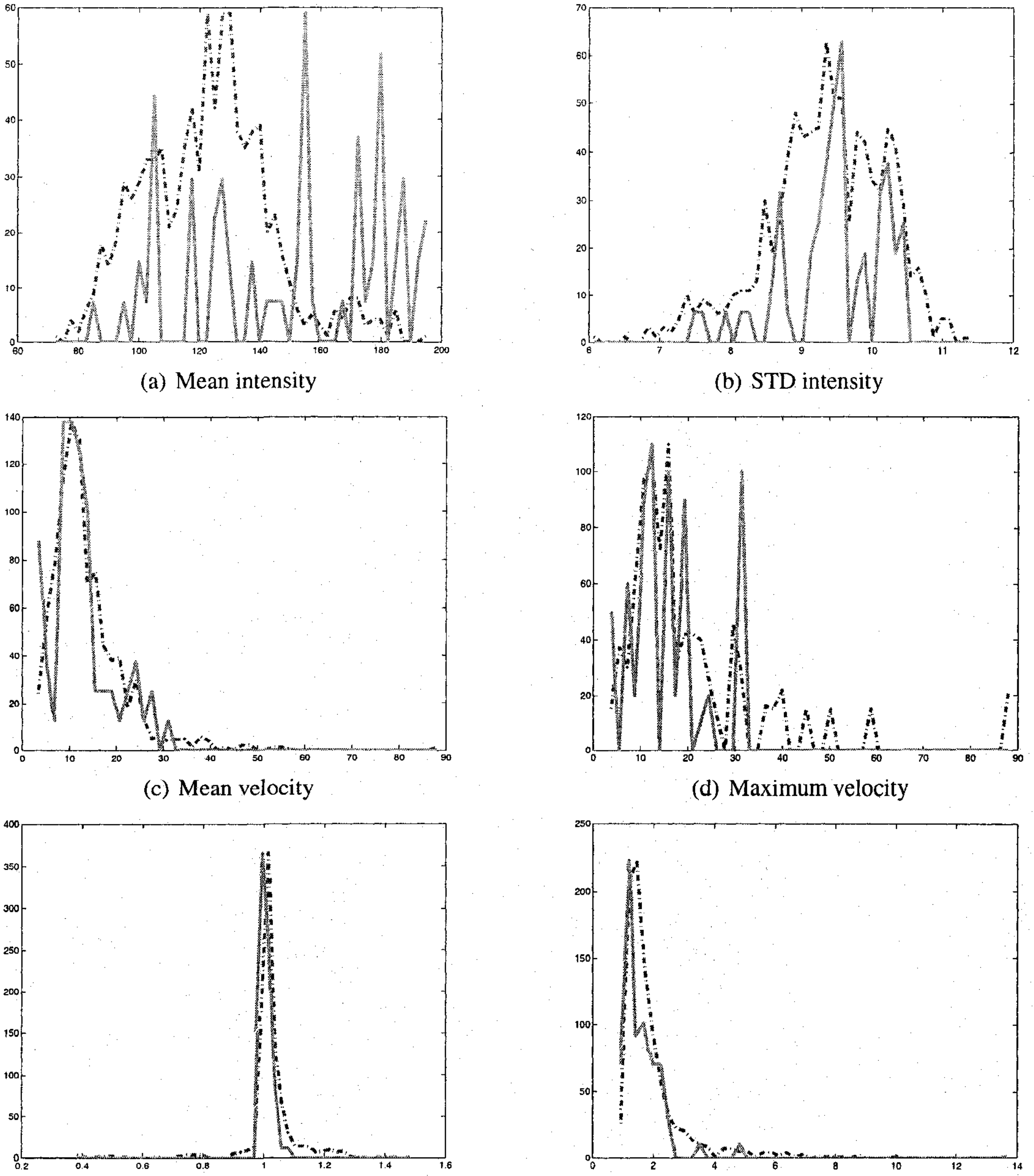
4.2.5 Applying The Learned Model

Decision Tree Learning

It would be very resource consuming to use all possible combinations of all nine parameters (Table 4.2 shows the mapping between Gestalt features and their parameters, Section 4.2.4 describes how parameters were calculated) and their values to determine the best parameters and their thresholds (*i.e.,* to use manual thresholding). A method was needed to find the optimal values for the chosen parameters automatically and produce the final detection results. Linear regression could be useful, since it is a fast and reliable method and previously it seemed to work with shape characteristics quite well [30]. However, it was not known if the relationship between the current set of parameters/features and the detection result was linear. Therefore, it was decided to use the machine learning method of decision trees, since it can handle nonlinear data prediction. In addition to handling nonlinear prediction, the structure of the decision trees (if-else statements, essentially) corresponds to the preferred design choice of the final region selection process - as an intuitive process of manual step-by-step selection. Thus, decision trees modeling was chosen, because they it was nonparametric, intuitive and robust. The algorithm of applying decision trees to large lump detection is shown in Figure 4.19.

Gestalt features	**Parameters**
Intensity	Mean intensity STD intensity
Motion	Mean velocity Max velocity
Shape	Roughness Compactness Eccentricity Solidity Extent

Table 4.2: The table shows correspondences between Gestalt features and their parameters.



(a) Mean intensity

(b) STD intensity

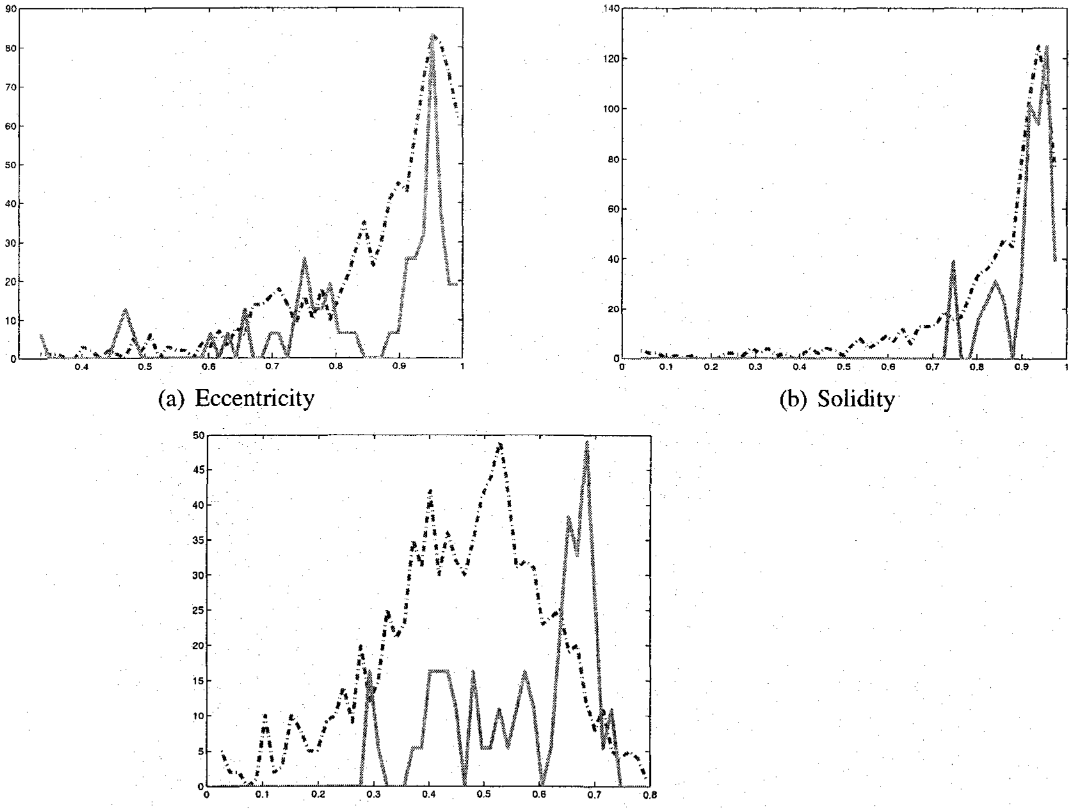
(c) Mean velocity

(d) Maximum velocity

(e) Roughness

(f) Compactness

Figure 4.13: Histogram plots of raw data for the following parameters: mean intensity, STD intensity, mean velocity, maximum velocity, roughness and compactness. Mean intensity seems to show better separation between two classes of lump candidates. A solid line corresponds to "good" lump candidates and a dotted line corresponds to "bad" lump candidates. See Table 4.2 for the list of parameters and their correspondence to Gestalt features.



(a) Eccentricity

(b) Solidity

(c) Extent

Figure 4.14: Histogram plots of raw data for the following parameters: eccentricity, solidity and extent. Extent seems to show better separation between two classes of lump candidates. A solid line corresponds to "good" lump candidates and a dotted line corresponds to "bad" lump candidates. See Table 4.2 for a list of parameters and their correspondence to Gestalt features.

A decision tree representation was first introduced by Raiffa and Schlaifer [32]. Any decision tree is a subclass of a general tree data structure. A generic tree is a hierarchical graph data structure with a set of linked nodes (see Figure 4.15). Each node that is higher in hierarchy is a parent node. Each node below the parent one is a child node. The top node in the hierarchy is known as the root and the bottom nodes are known as leaves. A decision tree is a binary tree. A binary tree is a tree, where each parent node has at most two children nodes (see Figure 4.16).

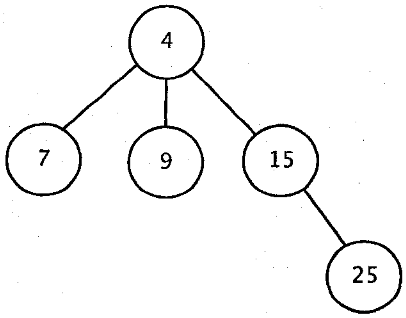


Figure 4.15: An example of a tree data structure. The top node is a root node that is also a top parent node in the hierarchy. Bottom nodes are its children nodes. Nodes 7, 9 and 25 are also leave nodes (*i.e.*, leaves), since they are the end-nodes of the tree.

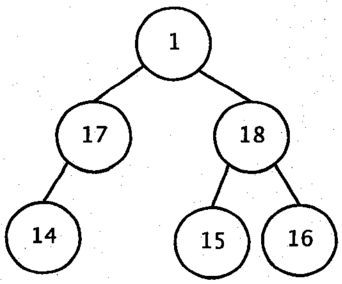


Figure 4.16: An example of a binary tree data structure. Each parent node (including root) can have at most two children nodes.

A classification decision tree was used in the proposed method (see Figure 4.17). The classification decision tree's leaves represented classification decision (1 for large lump detection and 0 for no detection) and links between nodes represented how Gestalt features' parameters were combined to lead to classification decisions. Each non-leaf node (*i.e.*, split) represented a decision rule that used a Gestalt feature's parameter and a value to decide what link should be followed next.

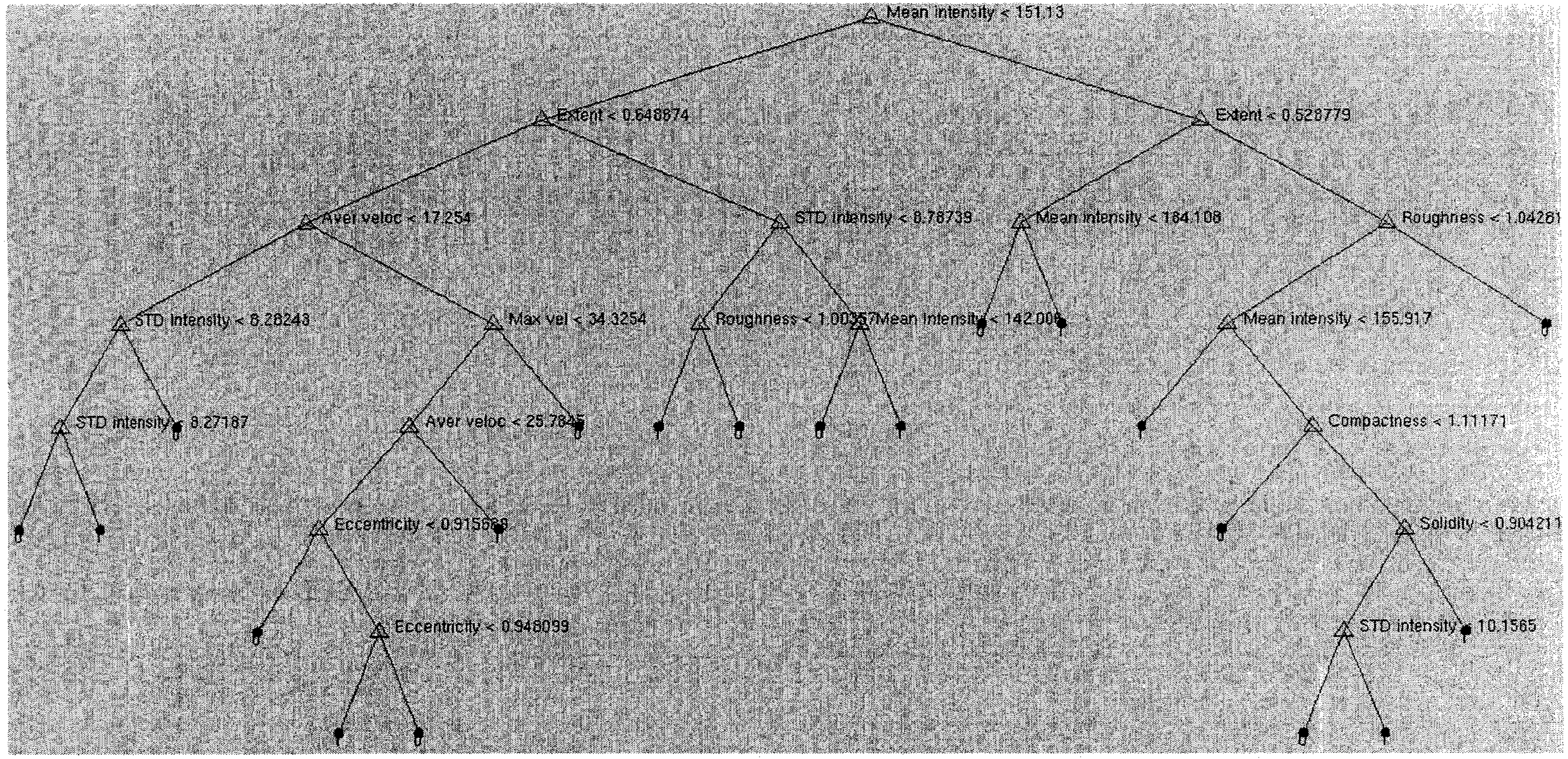


Figure 4.17: An example of a pruned decision tree used in the proposed method. This tree was created using a maximum deviance reduction splitting rule (see the next subsection for more detailed description) on all three Gestalt features (9 parameters). The growth of the tree was limited by forward pruning. Implementation was performed using decision tree Matlab [29] functions.

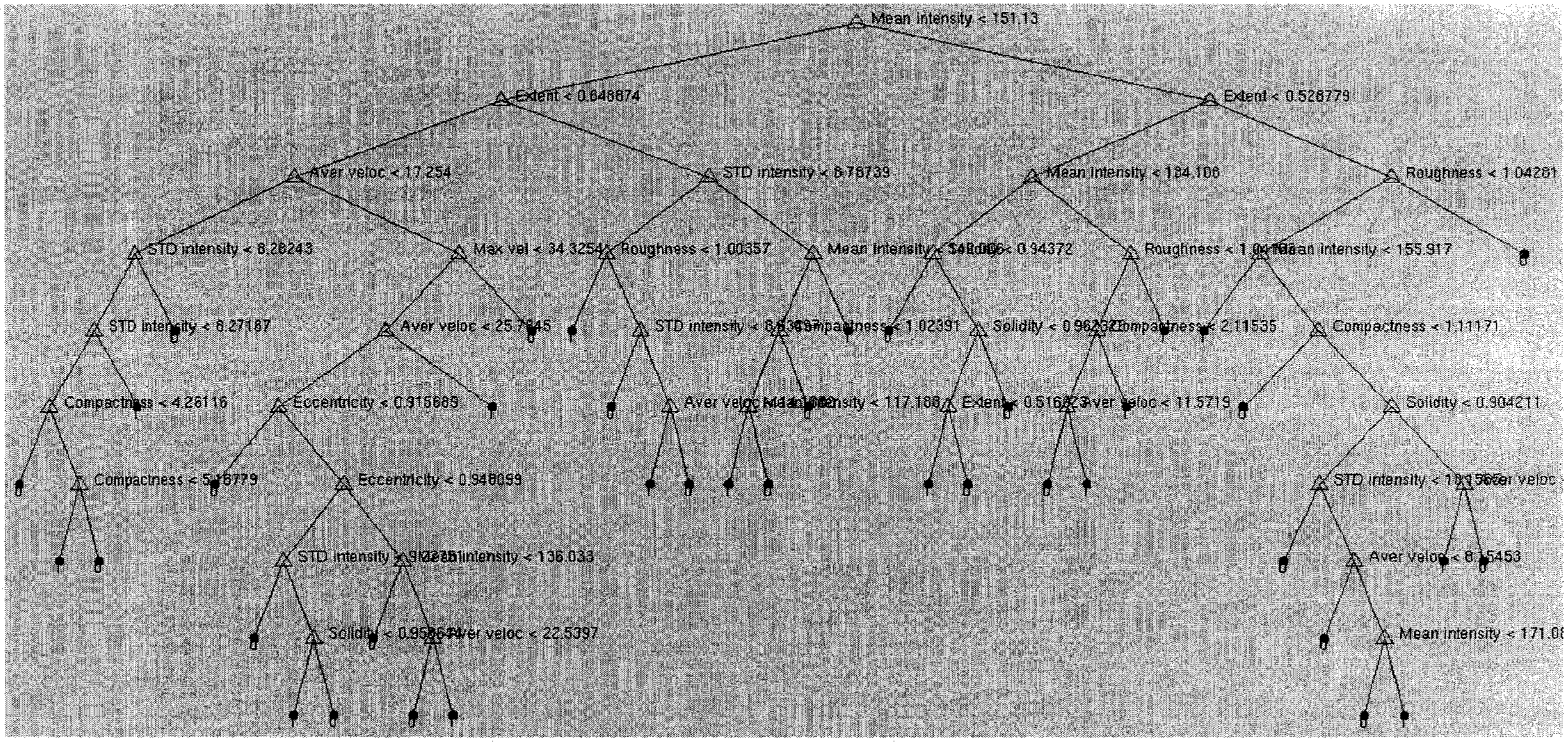


Figure 4.18: An example of an overfitted decision tree used in the proposed method. This tree was created using a maximum deviance reduction splitting rule on all three Gestalt features (9 parameters). The growth of the tree was not limited. Therefore, the overfitted tree had almost two times more terminal nodes (*i.e.*, leaves) than a pruned tree displayed in Figure 4.17

A way to optimize a decision tree is called pruning that implies downsizing a tree to avoid overfitting. Two methods could be used to prune a decision tree: forward pruning is used to stop the growth of a decision tree, and post-pruning is used cut the size of a tree after it has been created [3]. Forward pruning was used: the growth of a tree was limited by setting the minimum size of a group that can be split. The size of the group was chosen to be the minimum size at which all chosen parameters were used. The experimentally determined value was 15. Figure 4.18 displays an overfitted decision tree that has 39 terminal nodes (*i.e.,* leaves), and Figure 4.17) displays a forward-pruned tree that has its number of terminal nodes pruned to 20.

Decision trees use specific criteria to decide which parameter and value should be used. More specifically, the method finds the best splits for training data trying to minimize a specific criterion. Each split creates two groups of data - left and right, by analogy with the binary tree structure.

Algorithm *DetectLargeLumps*($C_n, GFD_n, GFD_{train}, DTM$)
Input: A list C_n of candidate regions' contours for frame n; Gestalt feature data GFD_n (in one of seven configurations: $i, m, s, i{+}m, i{+}s, s{+}m$, or $i{+}m{+}s$), which corresponds to C_n; Gestalt feature data GFD_{train} from the training sample; a decision tree splitting method DTM (GDI, MDR or Twoing rule).
Output: 1 if a large lump was detected among C_n, 0 if there was no detection.
(∗ Demonstrates how the detection module works ∗)

```
$T \leftarrow$ train a decision tree using $GFD_{train}$ and $DTM$
$Results \leftarrow$ apply $T$ to $GFD_n$
for $r$ is a detection result for a single candidate contour in $Results$
    do
        if $r = 1$
            $MajorAxisLength_r \leftarrow$ compute maximum dimension of $C_{n_r}$
            if $MajorAxisLength_r \geq 50\%$ apron feeder opening
                return $r$
return 0
```

Figure 4.19: An algorithm describing the detection process of the proposed method.

Splitting Criteria For Decision Trees

For each experiment three prominent (as per Teeuwsen *et al.* [46]) decision tree splitting criteria were used: Gini's diversity index, maximum deviance reduction and twoing rule.

Gini's diversity index (*i.e.,* GDI) is computed as follows:

$$\frac{n_l}{n}\left(1-\sum_{i=0}^{1}\left(\frac{l_i}{n_l}\right)^2\right)+\frac{n_r}{n}\left(1-\sum_{i=0}^{1}\left(\frac{r_i}{n_r}\right)^2\right) \tag{4.8}$$

where n is the total number of subjects in both groups, n_l and n_r is the number of subjects in, respectively, left and right groups, and l_i and r_i is the number of subjects of class i in the left or right group. Gini's diversity index measures the amount of impurity in both groups formed after splitting. The goal is to minimize the diversity index so that each group has less "impurities" (subjects from another group).

Let us imagine two groups for a specific parameter and a value. If the left group consists of 0,0,0,1 and the right group consists of 0,0,1,1,1,1 then GDI is going to be $\left(\frac{4}{10}\left(1-\left(\frac{3}{4}\right)^2-\left(\frac{1}{4}\right)^2\right)+\frac{6}{10}\left(1-\left(\frac{2}{6}\right)^2-\left(\frac{4}{6}\right)^2\right)\right)=0.42$. If the left group consists of 0,0,0,0 and the right group consists of 0,1,1,1,1,1 then GDI is going to be $\left(\frac{4}{10}\left(1-\left(\frac{4}{4}\right)^2-\left(\frac{0}{4}\right)^2\right)+\frac{6}{10}\left(1-\left(\frac{1}{6}\right)^2-\left(\frac{5}{6}\right)^2\right)\right)=0.17$. Thus, it can be seen that a lower GDI corresponds to a better separation of two classes.

Maximum deviance reduction (*i.e.,* MDR) minimizes the sum of variances within both groups:

$$\frac{1}{n_l}\sum_{j=1}^{n_l}(c_{lj}-\overline{c_l})^2+\frac{1}{n_r}\sum_{j=1}^{n_r}(c_{rj}-\overline{c_r})^2 \tag{4.9}$$

where n_l and n_r are the number of subjects in, respectively, left and right groups, c_j and c_k are j-th and k-th values of a corresponding group, and $\overline{c_l}=\frac{1}{n_l}\sum_{k=1}^{n_l}c_{lk}$, $\overline{c_r}=\frac{1}{n_r}\sum_{k=1}^{n_r}c_{rk}$. Deviances within two groups measure their homogeneity. For the first example above the sum of variances will be 0.41 and for the second example it will be 0.14. As it can be seen, lower deviance yields better class separation.

Twoing rule, similar to the other two criteria described above, checks if left and right groups are homogeneous. The twoing value is computed as follows:

$$\frac{n_l n_r}{n^2}\left(\sum_{i=0}^{1}\left|\frac{l_i}{n_l}-\frac{r_i}{n_r}\right|\right)^2 \tag{4.10}$$

where n is the total number of subjects in both groups, n_l and n_r are the number of subjects in, respectively, left and right groups, and l_i and r_i are the number of subjects of class i in the left or right group. For the examples above the twoing values will be 0.17 and 0.67, respectively. Twoing rule is a goodness measure rather than impurity measure. Therefore, high twoing value means that there is a good separation between two classes.

4.3 History Of The Proposed Method

Several ideas were tested that were not used directly in the proposed method. However, this information was reported for the completeness of the proposed research. The Gestalt principle of good continuation was tested and it was found that the principle of closure was a better alternative, since the good continuation principle was limited by its locality and uses only smoothness during edge selection. Using intensity slicing with shape learning, similarly to Lukas-Kanade motion estimation also did not seem to provide good results.

4.3.1 Good Continuation

The Gestalt principle of good continuation implies that contours, which belong to the same object tend to follow a continuous curve. It was assumed that continuous curves will be either lines or arcs. Therefore, the good continuation principle was measured by collinearity and cocircularity of the contours. Collinearity/cocircularity was assessed by fitting a line/circle to the points, which are close to the endpoints of candidate contours (see Figure 4.20). For each junction of branches inside a contour only two branches were joined based on the smallest value of collinearity/cocircularity. The value representing collinearity/cocircularity was set to be the average minimal distance from chosen points to the closest points of fitted line/circle.

It was noticed that using good continuation principle does not change the edge map much (see Figure 4.21). The reason seemed to be that the principle of good continuation is applied locally to two edges at a time, which does not allow for a

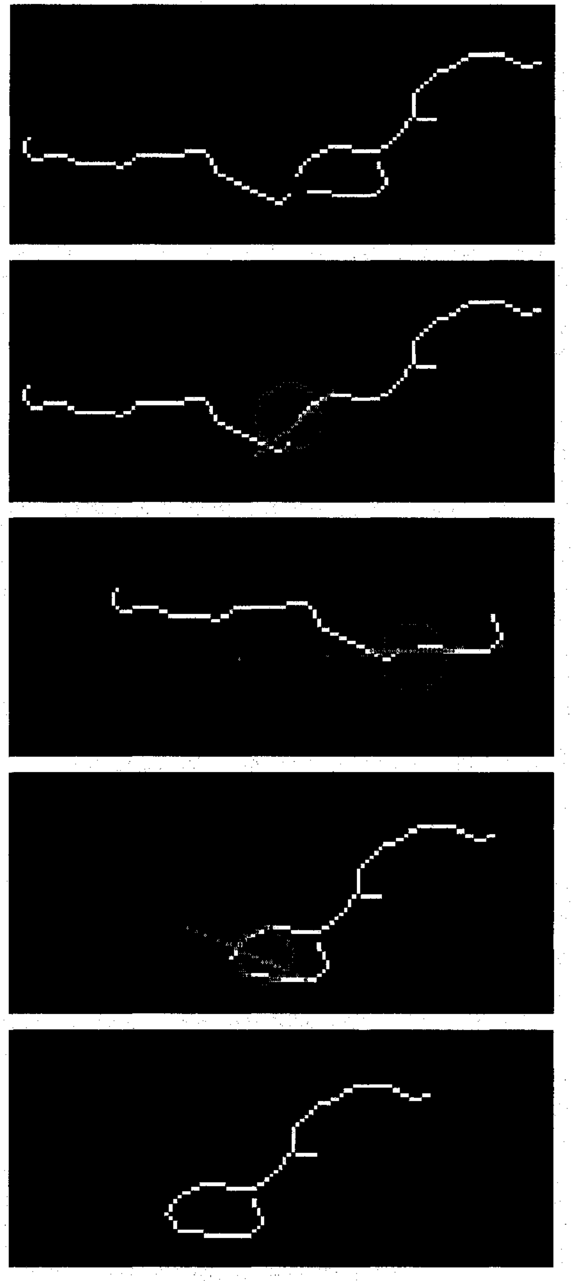


Figure 4.20: An example of the good continuation principle usage. Good continuation was expressed via collinearity and cocircularity by fitting, respectively, a line or a circle to points close the the endpoints of a pair of tested contours. The measure of collinearity/cocircularity was set to be the average distance from the chosen contour points to the closest points of the fitted line/circle. The topmost image displays a junction of three branches. Three rows below show all (three) possible combinations of edges with fitted lines/circles shown at the junctions. The image at the bottom is the combination that yields the smallest fitting error value.

higher level view of edges.

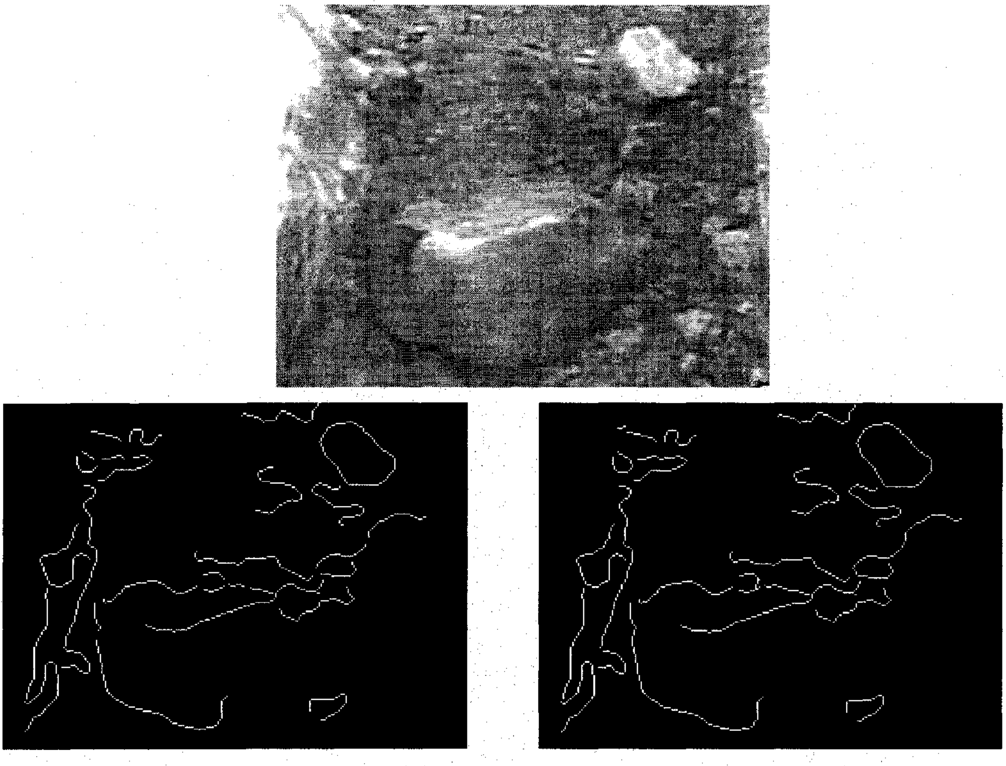


Figure 4.21: The left bottom image contains an edge map obtained after applying a Canny edge detector. The right bottom image contains the result of applying the good continuation principle to the left bottom image. Notice that the locality of the good contimnuation principle does not allow it to make a significant improvement.

Thus, the main reason that the good continuation principle failed to have a significant impact on the edge images seemed to be the local nature of the principle. Therefore the proposed method turned to a more global implementation of the good continuation principle, which, in addition to smoothness, also takes edges' shape into account. The new principle was the principle of closure. The principle of closure implied that an object was still seen as a whole object even if some parts were missing, because previous knowledge about the object allowed to fill the gaps in. In the proposed method the principle of closure was realized via filtering configurations of connected edges using their intensity, motion and shape information.

4.3.2 Slicing

Recently, intensity slicing of images followed by region filtering based on learned shape characteristics was shown to provide good results on oil sand images [30]. Mukherjee *et al.* [30] used such shape characteristics as eccentricity, solidity, and extent to select the best shapes among those that were created after slicing. The above mentioned method was applied to large lump detection. Segmentation of large lumps based on linear learning of shape features as described by Mukherjee *et al.* [30] did not seem to provide very good results when used by itself (see Figure 4.22).

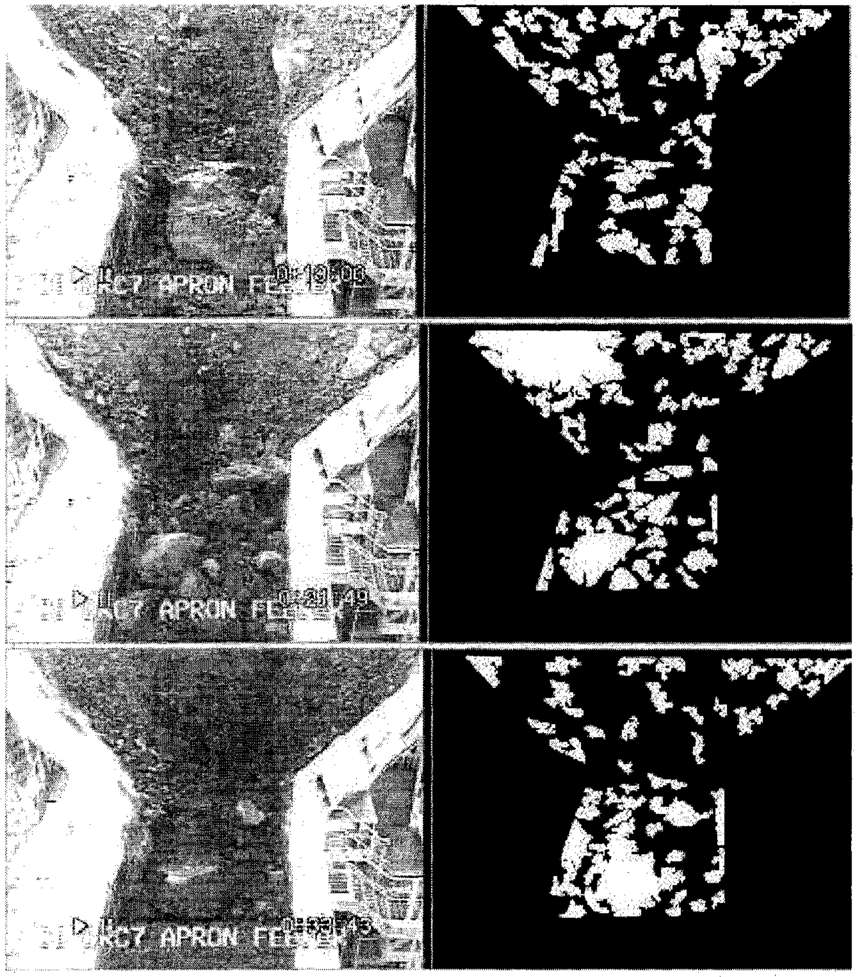


Figure 4.22: Using slicing method to segment large lumps. The large lump was not detected in row one. Also, a false large lump was found in a row two image.

4.3.3 Motion

Lukas-Kanade motion estimation was tested to determine the velocity vectors of large lumps. However, the velocity vector estimation proved to be very prone to numerical errors, and problematic because of the uneven illumination and texture, and not very high relative depth for large lumps and other material. Thus, as it could be seen from Figure 4.23 pixels-based motion segmentation did not seem to provide good results.

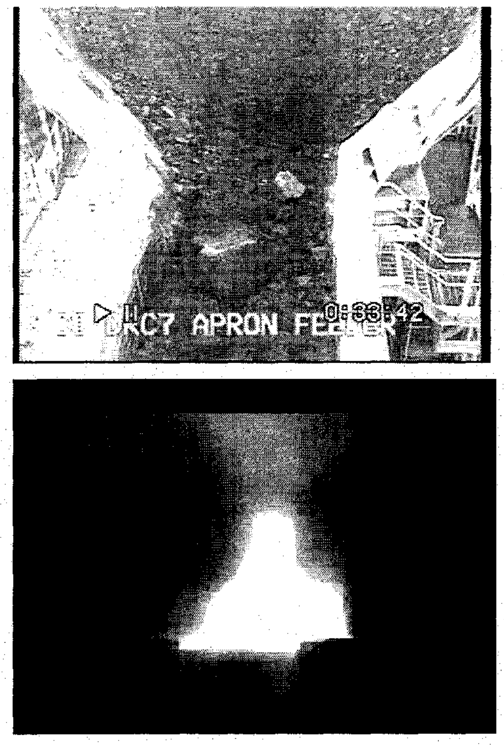


Figure 4.23: Using Lukas-Kanade motion estimation to segment large lumps. The method seemed to be prone to numerical errors.

4.4 Summary

In the current chapter the proposed method of large lump detection was described. The proposed method consisted of four major steps: edge detection, matching of moving edges, creation of candidate large lumps and detecting large lumps. The most important point of the method was that it used Gestalt features of intensity, motion and shape together with machine learning to detect large lumps. The next chapter will review experimental data and performance measures of the proposed method. It will also discuss statistical methods that were used to test the hypothesis.

Chapter 5

Experimental Design

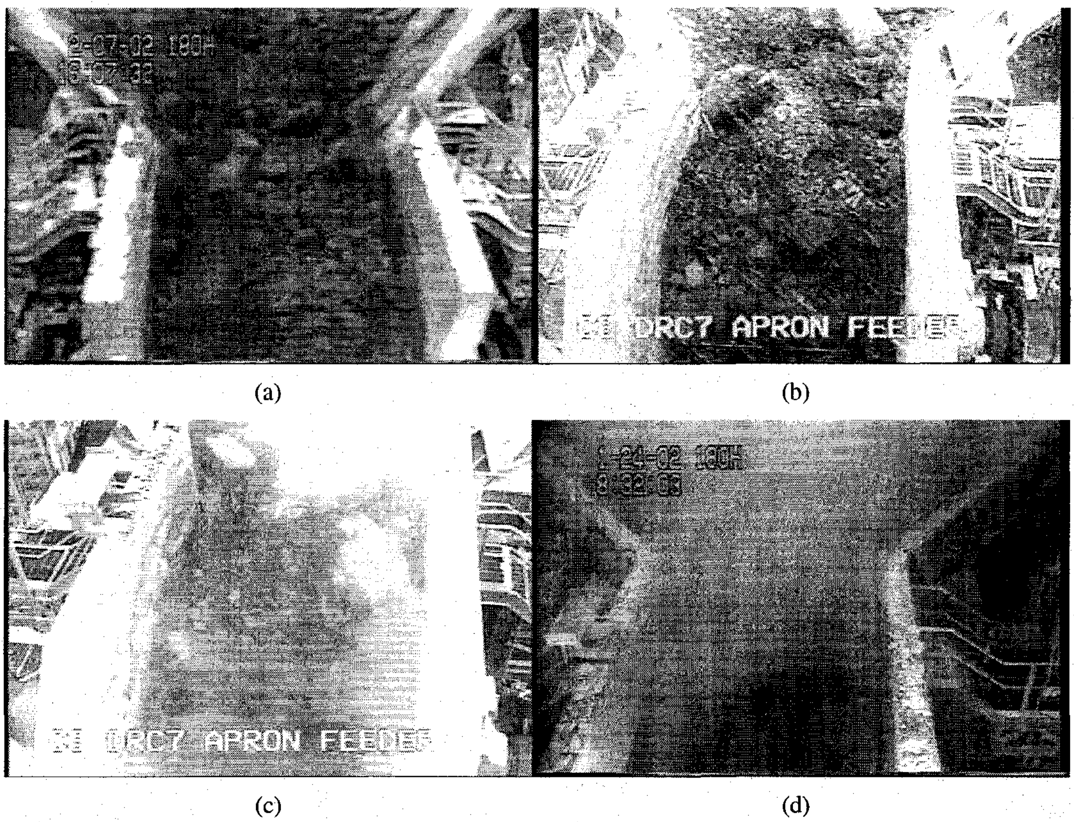
This chapter describes experimental configurations of Gestalt features and input data used for experiments. Statistical methods and performance measures used to evaluate the proposed method are also discussed.

5.1 Experimental Data

The main dataset used in experiments consisted of 2446 images (*i.e.*, frames digitized from large lump video) that contained 46 large lump events.

5.1.1 The Main Dataset

Initial data was provided by Dr. Ron Kube of Syncrude Canada Ltd. Five video tapes were reviewed. Four video tapes contained approximately twelve days of video each (one frame per second) and one video tape contained six days of video (four to five frames per second). Most of the videos had poor quality caused by poor lighting conditions (video that was taped at night), snow, and large amounts of steam (see Figure 5.1). Not all tapes had actual large lumps in them. Thus, much of video data was not very useful for the proposed method testing. Considering that approximately 41 minutes of large lump video was chosen from the SVHS Tape 6. Initial SVHS data was recorded at one frame per second (approximately 425 lines of horizontal resolution). Therefore, the chosen sample consisted of 2446 frames. Chosen data was transfered to a digital video (a DV) tape at the rate of thirty frames per second creating approximately one minute and twenty two seconds of large lumps video. DV data was digitized using WinDV freeware [43] into a single DV Type 2 AVI file. All frames of the file were captured by ImageGrab 3.0 freeware [42] as BMP 720x480 images of size 1mb each.



(a)

(b)

(c)

(d)

Figure 5.1: Examples of problems with large lump images: uneven lighting in 5.1(a), snow in 5.1(b), steam and lighting problems in 5.1(c) and 5.1(d).

Large Lumps

All images were visually analyzed and it was determined that they contain 46 large lump events. A *large lump* was defined as a lump of oil sand that had its maximum dimension equal to or larger than one-half the chute's width. Every *large lump event* had to consist of at least two frames, where each frame contained one or more large lumps. Thus, the largest oil sand lump in each large lump event's frame had to be at least half of the chute's width in lump's maximum dimension (see Figure 5.2 for an example).

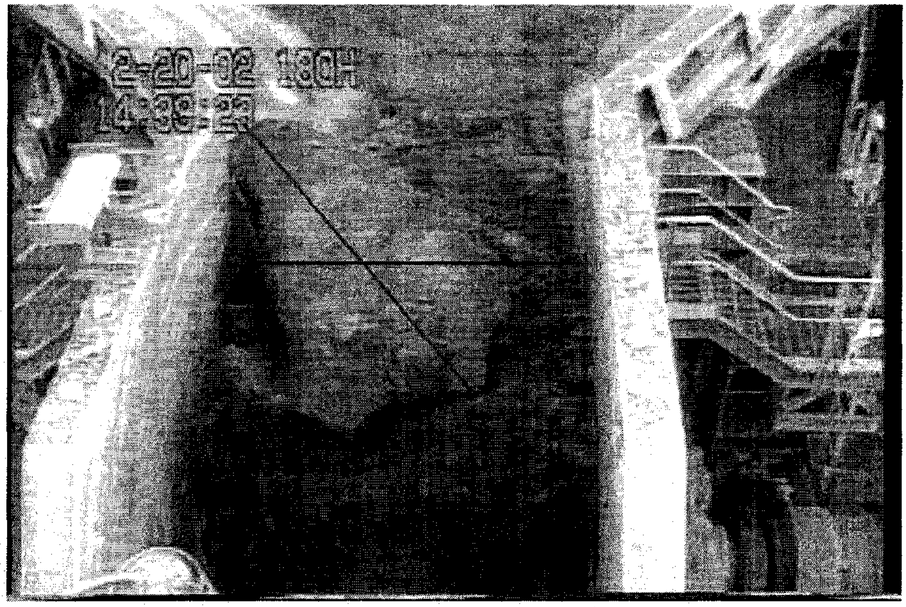


Figure 5.2: Visual evaluation of large lumps. The observer labeled a lump as a large lump if it were more than a half of chute's width (horizontal line) in its maximum dimension (tilted line). The chute's width changes when moving up or down the image, because of the perspective projection of the camera. Therefore, scale normalization was performed (it is described in "Velocity Computation" subsection of Section 4.2.2).

5.1.2 The Training Sample

The training sample was a randomly chosen subsample of the main dataset that was later used to train (*i.e.,* create) a decision tree model for each configuration of Gestalt features. It consisted of 40 images corresponding to 20 two-frame inputs (*i.e.,* tuples) to the main algorithm. Each tuple was selected randomly from a 2446-image sample. One half of the images contained large lumps and the other half did not (i.e. there were only 10 tuples containing large lumps).

5.2 Experimental Configurations

Thus, with the selection method in place, seven experiments were designed. The experiments would use following combinations of three Gestalt features as input data to segment large lumps from the background. The combinations were: intensity (*i.e.,* i), motion (*i.e.,* m), shape (*i.e.,* s), intensity+motion (*i.e.,* $i + m$), intensity+shape (*i.e.,* $i + s$), motion+shape (*i.e.,* $m + s$), and intensity+motion+shape (*i.e.*, $i+m+s$). Each configuration was tested three times (each time corresponded to a different splitting criterion of the decision tree) and performance measures (see Section 5.3) were computed for each trial.

5.3 Performance Measures

The performance measures that were used to assess the experimental results were accuracy, precision, recall (*i.e.*, true positive rate) and false positive rate. They were computed as follows:

$$accuracy = \frac{TP+TN}{TP+FP+TN+FN} = \frac{TP+TN}{P+N} \tag{5.1}$$

$$precision = \frac{TP}{TP+FP} \tag{5.2}$$

$$recall = \frac{TP}{TP+FN} \tag{5.3}$$

$$FP_{rate} = \frac{FP}{FP+TN} \tag{5.4}$$

where TP are true positives (*i.e.*, situations when a lump was present and it was detected), TN are true negatives (a lump was not present and it was not detected), FP are false positives (a lump was not present and it was detected), FN are false negatives (a lump was present and it was not detected), P are all detections ($TP+FP$), and N are all situations were there were no large lumps detected ($TN+FN$). With respect to the application precision could be considered the most important measure, since the primary concern of a large lump detection system would be to minimize the number of false alarms (i.e. FPs). However, a good object recognition system should have a balance between its precision and recall. With respect to research not only precision but also accuracy should be the main performance measure characteristics, since their combination is traditionally used to measure performance of methods.

5.4 Statistical Analysis Methods

5.4.1 Confidence Intervals

Each configuration of Gestalt features was tested three times (each time corresponded to a different splitting criterion of the decision tree) and a 95% confidence interval was computed for three trials. 95% confidence intervals were used to determine if the means of results for each configuration and Gestalt group come from the same population.

5.4.2 T-Test

Student's t-test was used to compare perfomance of large lump detection when using one, two and three Gestalt features. Student's t-test indicated if differences between two samples were due to a chance (a null hypothesis[1] was accepted for a t-test's p-value greater or equal to a statistical significance of 0.05) or not (a null hypothesis was rejected for a significance value lower than 0.05). Thus, a rejected null hypothesis implied that two samples are statistically different. It was assumed that data in samples was normally distributed and group variances were equal. The samples were considered to be dependent, since they used the same input data with different Gestalt features applied to it.

One, two, and three-feature results were grouped into three corresponding groups that were compared to each other using a t-test. One-feature group contained "intensity", "motion" and "shape" results, two-feature group contained "intensity+motion", "motion+shape" and "intensity+shape" results, and three-feature group contained "intensity+motion+shape" results. As a result, the size of one and two-feature samples that had three times more values than a three-feature sample. Therefore, the results of "intensity+motion+shape" configuration were used three times to match the size of one and two-feature samples, since the size of dependent samples should be equal for the t-test.

[1] A hypothesis of no differences between two groups. It is presumed to be true until (statistically) proved otherwise.

5.4.3 Linear Regression

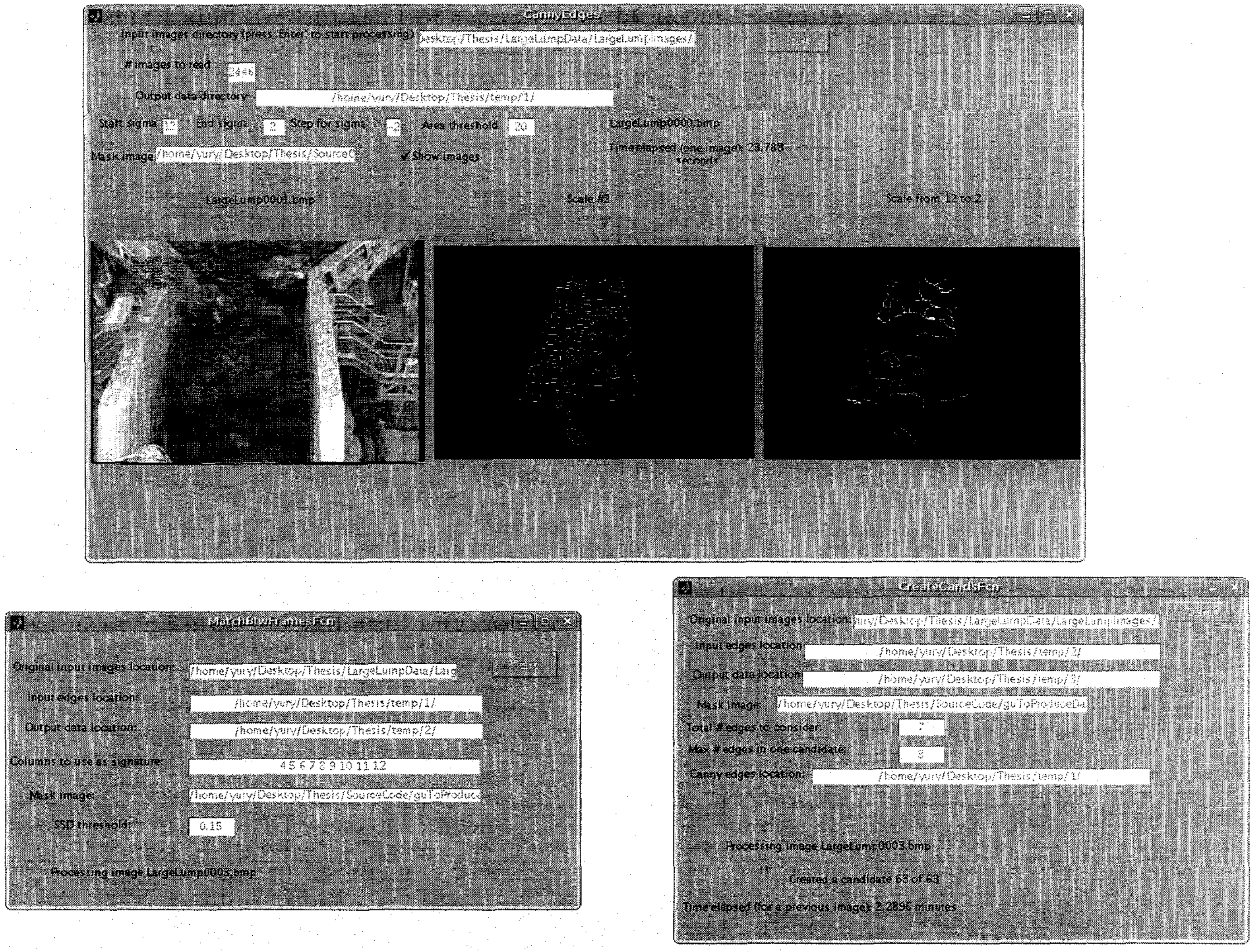
Linear regression helped to determine if performance measures' values monotonically increased/decreased with higher number of Gestalt features for groups that were statistically different. Linear regression was used to supply statistically significant differences (results of Student's t-test) with directional information, since it did not make sense to evaluate direction for groups that were not statistically different. Thus, t-test was used first to determine significance. If statistical significance was found, then linear regression was applied to evaluate the direction of differences between the groups.

In the proposed linear regression analysis the relationship between each performance measure value and the number of Gestalt features was modeled using the formula of a first-degree polynomial: $y = \beta PM + \beta_0$, where y is a label for a number of Gestalt features that corresponds to a performance measure value PM, and β_0/β are coefficients to be found. Fitting of the linear model to data was performed using linear least squares' formula: $\begin{bmatrix} \beta \\ \beta_0 \end{bmatrix} = (X^T X)^{-1} X^T y$, where $X = \begin{bmatrix} PM_1 & 1 \\ \dots & \dots \\ \dots & \dots \\ PM_n & 1 \end{bmatrix}$ and y is the number of Gestalt features used to obtain a corresponding PM. For example, $y = \begin{bmatrix} 1 \\ \dots \\ 2 \\ \dots \\ \dots \\ 3 \end{bmatrix}$.

Computed β and β_0 were used to fit a linear model (graphically expressed as a line) for $y = 1, 2$, and 3 to make a decision whether the performance measure PM monotonically increased/decreased with increasing number of Gestalt features (y in this case). Function y was assumed to be linear just for this part of the statistical analysis.

5.5 Implementation Details

The application based on the proposed method was implemented in Matlab [29] using Image Processing and Statistics Toolboxes. Statistics Toolbox was also used to compute linear regression and t-test results. Matlab's module GUIDE was used to create GUIs (see Figure 5.3). GUIs were created for the following modules: edge detection, motion matching of edges and candidate region creation.



Input images directory (press 'Enter' to start processing) Desktop/Thesis/LargeLumpData/LargeLumpImages/

images to read 2446

Output data directory /home/yury/Desktop/Thesis/temp/1/

Start sigma 12 End sigma 2 Step for sigma -2 Area threshold 20

LargeLump0000.bmp

Mask image /home/yury/Desktop/Thesis/SourceC

✓ Show images

Time elapsed (one image): 23.788 seconds

Original input images location: /home/yury/Desktop/Thesis/LargeLumpData/Larg

Input edges location: /home/yury/Desktop/Thesis/temp/1/

Output data location: /home/yury/Desktop/Thesis/temp/2/

Columns to use as signature: 4 5 6 7 8 9 10 11 12

Mask image: /home/yury/Desktop/Thesis/SourceCode/guToProduc

SSD threshold: 0.15

Processing image LargeLump0005.bmp

Original input images location: yury/Desktop/Thesis/LargeLumpData/LargeLumpImages/

Input edges location /home/yury/Desktop/Thesis/temp/2/

Output data location /home/yury/Desktop/Thesis/temp/3/

Mask image: /home/yury/Desktop/Thesis/SourceCode/guToProduceDa

Total #edges to consider: 7

Max # edges in one candidate: 8

Canny edges location: /home/yury/Desktop/Thesis/temp/1/

Processing image LargeLump0003.bmp

Created a candidate 63 of 63

Time elapsed (for a previous image): 2.2896 minutes

Figure 5.3: .GUI screenshots for edge detection, motion matching and candidate region creation modules.

5.6 Summary

The chapter discussed experimental data, result evaluation measures, and statistical methods that were used to validate the hypothesis. The next chapter will focus of the review of the proposed method's results.

Chapter 6

Results and Discussion

It was statistically confirmed that using multiple Gestalt features improved overall performance of large lump detection. Student's t-test statistics proved that large lump detection results for one, two and three-feature groups were significantly different from each other with respect to three out of four main performance measures (one performance measure did not show any difference between groups). For groups that were statistically different from each other, linear regression showed that detection performance improved with increased number of Gestalt features. Means, 95% confidence intervals, and analysis of visual output were consistent with statistical results.

6.1 Results

There were statistically significant differences detected between results obtained using one, two and three Gestalt features for the following performance measures: accuracy, precision and false positive rate (see Table 6.1). The recall performance measure was statistically the same for any number of Gestalt features (see Table 6.1). Training and testing errors also showed significant differences with an increasing number of Gestalt features. There were no statistically significant difference between one- and two-feature results for the training error.

Linear regression results assisted in determination of the direction of statistically significant differences between groups with different number of Gestalt features. Linear regression model suggested that the performance of the proposed method increased with increasing number of Gestalt features (see Figure 6.1) for those performance measures that showed statistically significant differences between groups.[1] Thus, t-test and linear regression results show that there was a statistically significant overall tendency for increased performance, when the number of Gestalt features increased.

Figures[2] 6.3(a), 6.3(b), and 6.3(d)) help to illustrate that three-feature detection had higher performance than one or two-feature detection in terms of 95% confidence intervals. Figures A.9, A.10, A.11, and A.12 show some visual detection output, which is consistent with numerical results.

The combination of all three Gestalt features had the smallest number of false detections and the highest mean number of correctly identified images that did not contain large lumps. Also, the combination of intensity, motion and shape Gestalt features appeared to be the best feature combination, because it had the lowest variability across the splitting criteria and the greatest separation from other feature combinations. The "i+m+s" combination had better performance measure values for the Gini's diversity index decision tree splitting rule. GDI appeared to be a better choice compared to other decision tree splitting criteria. GDI (unlike MDR - see Formula 4.9) took the group size into account, and (unlike Twoing rule - see Formula 4.10) did not favour equal group size.

Results in the Appendix provide more detailed data for each Gestalt feature used in experiments. Figures A.1 and A.2 contain mean (from three decision tree splitting criteria) results for each Gestalt feature combination used in the experiments. Figures A.3-A.8 contain results displayed separately for each of three decision tree splitting criteria: Twoing, maximum deviance reduction and Gini's diversity index. Tables A.1, A.2, and A.3 contain numeric results obtained using Twoing, maximum deviance reduction and Gini's diversity index splitting criteria.

[1] Directional information provided by linear regression would not make sense if there were no statistically significant differences shown by t-test.

[2] In all the figures the abbreviations should be understood as follows: "i" - intensity, "m" - motion, "s" - shape, "i+m" - intensity and motion, "i+s" - intensity and shape, "m+s" - motion and shape, intensity, "i+m+s" - motion and shape.

Traning error			
First group	**Second group**	**p-value**	H_0 ($\mu_{first} = \mu_{second}$) **rejected?**
1 Gestalt feature	2 Gestalt features	0.07	No
1 Gestalt feature	3 Gestalt features	0.00	Yes
2 Gestalt features	3 Gestalt features	0.01	Yes

Testing error			
First group	**Second group**	**p-value**	H_0 ($\mu_{first} = \mu_{second}$) **rejected?**
1 Gestalt feature	2 Gestalt features	0.00	Yes
1 Gestalt feature	3 Gestalt features	0.00	Yes
2 Gestalt features	3 Gestalt features	0.00	Yes

Accuracy			
First group	**Second group**	**p-value**	H_0 ($\mu_{first} = \mu_{second}$) **rejected?**
1 Gestalt feature	2 Gestalt features	0.00	Yes
1 Gestalt feature	3 Gestalt features	0.00	Yes
2 Gestalt features	3 Gestalt features	0.00	Yes

Precision			
First group	**Second group**	**p-value**	H_0 ($\mu_{first} = \mu_{second}$) **rejected?**
1 Gestalt feature	2 Gestalt features	0.03	Yes
1 Gestalt feature	3 Gestalt features	0.00	Yes
2 Gestalt features	3 Gestalt features	0.00	Yes

Recall			
First group	**Second group**	**p-value**	H_0 ($\mu_{first} = \mu_{second}$) **rejected?**
1 Gestalt feature	2 Gestalt features	0.26	No
1 Gestalt feature	3 Gestalt features	0.19	No
2 Gestalt features	3 Gestalt features	0.51	No

False positive rate			
First group	**Second group**	**p-value**	H_0 ($\mu_{first} = \mu_{second}$) **rejected?**
1 Gestalt feature	2 Gestalt features	0.01	Yes
1 Gestalt feature	3 Gestalt features	0.00	Yes
2 Gestalt features	3 Gestalt features	0.01	Yes

Table 6.1: Student's t-test results for the accuracy performance measure. Three out of four main performance measures (accuracy, precision and false positive rate) showed statistically significant differences between groups with different number of Gestalt features. The direction of these differences was shown by linear regression in Figure 6.1. The recall performance measure did not show any differences between groups of Gestalt features.

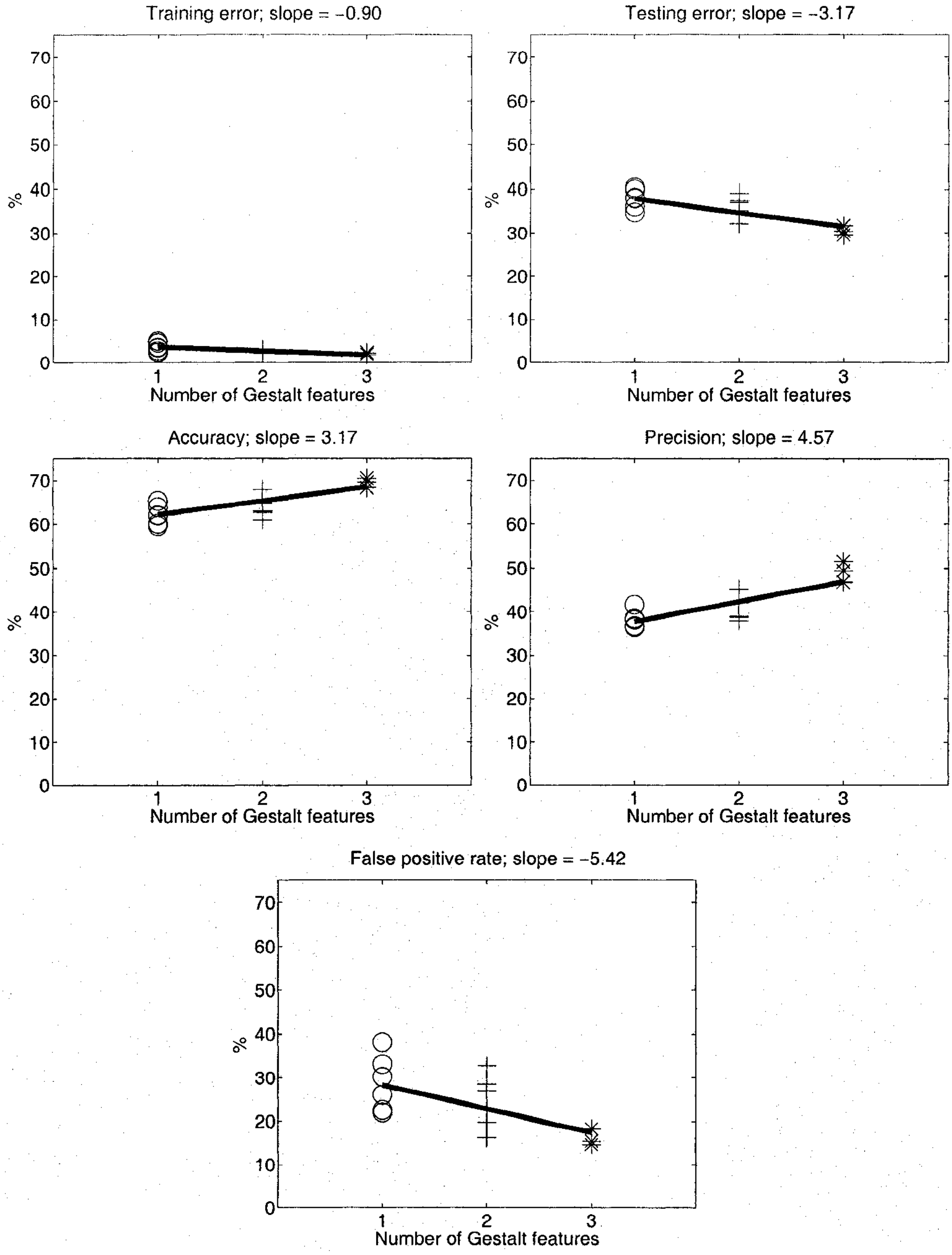
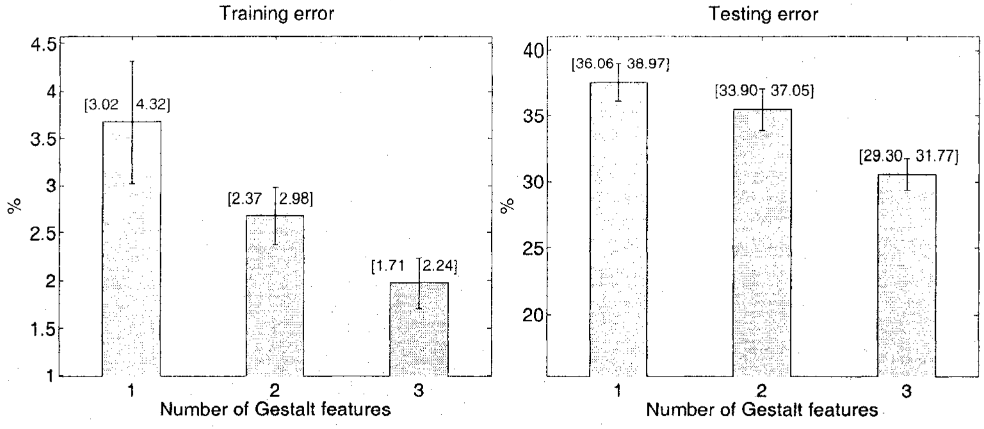


Figure 6.1: Linear regression model (described in Section 5.4.3) is represented here by a line. The model provides evidence if performance measures monotonically increase/decrease as the number of statistically different Gestalt features increases. The recall performance measure did not show any statistically significant differences between groups and, therefore, the direction of non-existent differences was not evaluated. The performance of the main measures (accuracy, precision and false positive rate) increased with increasing number of Gestalt features. The rate of increase/decrease is indicated by the slope of a line. The figures are shown on the same scale to make slopes comparable.



(a) (b)

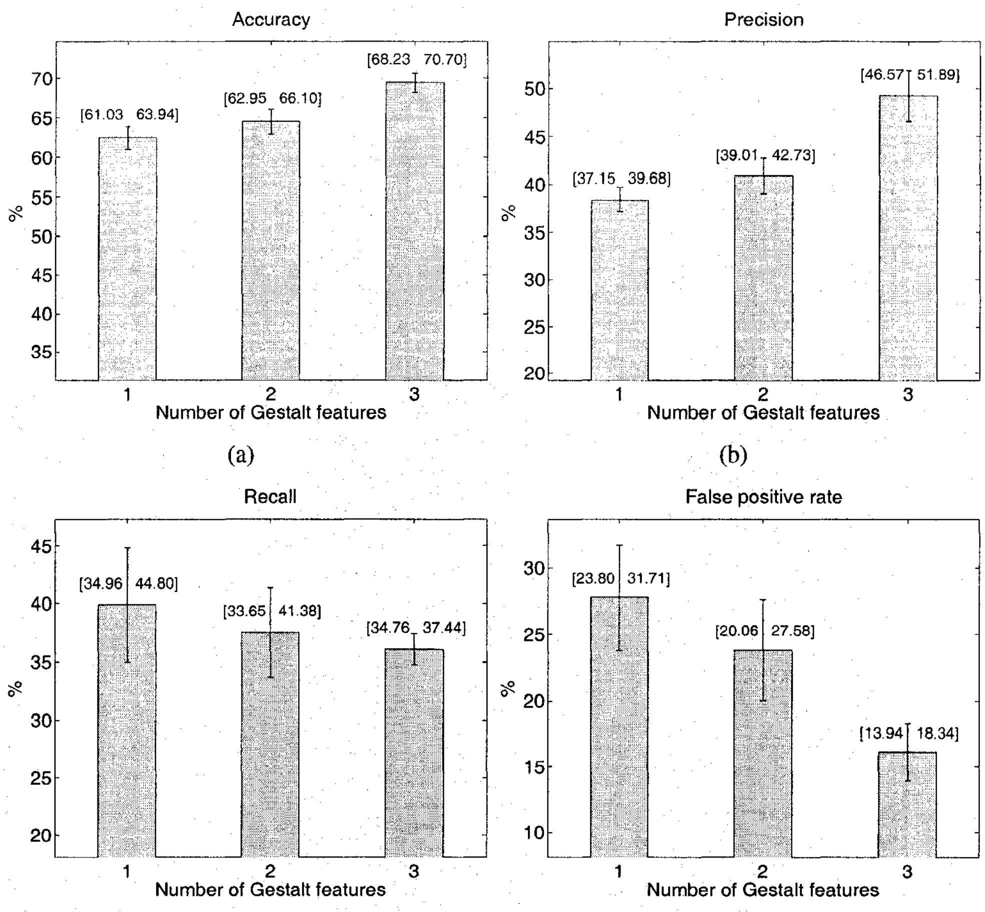
Figure 6.2: Mean training and testing error, as standard machine learning performance measures, for a different number of Gestalt features (1 feature: i, m, or s, 2 features: $i + m, m + s$, or $i + s$, 3 features: $i + m + s$). Training error is $1 - accuracy$ for the training group prediction, and testing error is $1 - accuracy$ for the test group prediction. Both errors decrease with the increasing number of Gestalt features. More deatailed results are available in Appendix.

6.2 Discussion

A greater number of Gestalt features produced statistically better results, as per t-test and linear regression comparison of one, two and three-feature groups. Training sample detection results from Figures A.9 and A.10 did not show much difference between one- and two-feature results. However, two-feature configurations in randomly selected results from Figures A.11 and A.12 seemed to detect better lump candidates than one-feature configurations.

There was no statistically significant separation between groups for a recall performance measure. The absence of separation could imply a higher number of missed detections for more Gestalt features (probably due to higher selectiveness of two or three-feature detection), even though the most of overall performance increased with higher number of Gestalt features.

Configurations that contained a shape Gestalt feature had a high variance across different decision tree splitting criteria and could be responsible for the lower difference between one-feature results and two-feature results. Most variability seemed to have come from MDR decision tree splitting rule.



(a) (b)

(c) (d)

Figure 6.3: Mean results for each group of Gestalt features. There is an evidence that the 3-feature result and 1/2-feature results are really two different groups, since the 95% confidence intervals (error bars, also numbers in brackets) do not overlap. Each group consists of corresponding performance measure values (accuracy, percision, recall, or false positive rate) obtained for all Gestalt features of that group and for all three decision tree splitting criteria. The breakdown of results by configurations of cues as well as decision tree splitting creteria is available in Appendix.

Twoing and Gini's diversity index criteria seemed to have the same or very similar results for all combinations of Gestalt features[3]. However, the maximum deviance reduction rule yielded quite different results than other two decision tree splitting criteria[4]. For example, there was approximately 4-16% change in values for the performance measures of the shape Gestalt feature and a combination of motion and shape (Tables A.1, A.2, and A.3). Recall and false positive rate for the shape Gestalt showed the highest increase for the maximum deviance reduction rule (of about 16%). The same performance measures for the shape and motion Gestalt combination showed an increase of about 9%, which is also quite high. Accuracy and precision dropped by approximately 3-6%. Such deviations appeared to be caused by a higher number of correct and incorrect detections (*i.e.,* true and false positives) and a lower number of correctly identified non-large lump images (*i.e.,* true negatives) for the maximum deviance reduction than for GDI or Twoing rules.

The number of incorrect detections (*i.e.,* false positives) was higher (compared to an increase in the number of correct detections) for the shape partial gestalt and a combination of motion and shape partial gestalts, when using maximum deviance reduction (*i.e.,* MDR) splitting criterion. Thus, MDR yielded significantly higher number of true and false detections and lower number of true negatives for "s" and "m+s" compared to corresponding numbers of GDI and Twoing rules. That did not seem to be a good result, since the increase in the number of true positives was much lower than increase in the number of false positives. Furthermore, a following question was raised: why did MDR results for shape and shape and motion noticeably differ from corresponding results of GDI and Twoing rule, while other parameter configurations did not have such variation?

[3] That makes sense, since their formulas (4.8 and 4.10) are somewhat similar. The main difference is that the Twoing rule measures the purity of two subsamples, and the Gini's diversity index measures the impurity, thus making one rule a conceptual inverse of the other. Also, Twoing rule seems to encourage equal size of splitted groups.

[4] The maximum deviance reduction (*i.e.,* MDR) rule produced more detections (both true and false ones) than the other two rules. The reason could be that, unlike other two rules, MDR did not take into account the size of the "good" large lumps group (which was significantly smaller than the size of the "bad" large lumps group), and the influence of the "good" group on the detection result was therefore exaggerated. As a result, the MDR's output had more detections than the output of the other two methods.

The shape partial gestalt seemed to be very sensitive to the splitting rule selection. The main difference between MDR and the other two rules was that the error measures of GDI and Twoing rule were computed using the size of classification groups, which was not the case with MDR. In the training sample the number of non-large lump subjects (*i.e.,* "bad" lumps) was substantially higher than the number of large lump subjects (*i.e.,* "good" lumps), which was also true for the large oil sand lump problem in general. Therefore, the large size of the "bad" group would give it more influence in the classification tree creation for GDI and Twoing rule.

In MDR the influence of both groups was equal, thus making the influence of "bad" lumps less powerful and the influence of "good" lumps more powerful in large lump classification, compared to GDI and Twoing rule results. Many true negatives seemed to have been labeled as false positives, since the classification power of "bad" large lumps was decreased in MDR. Many false negatives seemed to have been labeled as true positives, since the classification power of "good" large lumps was increased in MDR. Other Gestalt features had a smaller response to not using classification group sizes as error weights in MDR. This, along with higher sensitivity of the shape and shape and motion Gestalt features to MDR decision tree splitting rule, could imply that the shape Gestalt feature was mostly useful at selecting large lump events, and does not favour non-large lump data.

6.3 Summary

A statistically significant difference (for three out of four performance measures) was found in large lump detection performance for varying number of Gestalt features. The difference was also confirmed by observing selected visual detection output. Detection performance (based on the majority of most important performance measures) increased with larger number of Gestalt features. There was no statistically significant difference between feature configurations for the recall performance measure. The absence of differences could be caused by a higher number of false detections (that would nullify the influence of increased true detections for the recall performance measure). Another reason could be a high variability of shape-containing configurations. The reason of the higher variability of the shape Gestalt feature seemed be a greater number of true and false detections when using MDR rule.

The difference between one-feature results and two-feature results did not seem to be as large as between one/two-feature results and three-feature results. The reason could be a higher variability of results that contained a Gestalt feature of shape. The next chapter will draw conclusions about current work and will outline future directions of research.

Chapter 7

Conclusions

7.1 Contributions

This thesis presents a novel object detection method that uses multiple Gestalt features integrally with the machine learning framework to detect large lumps of oil sand. To my knowledge there is a limited body of research in computer vision on using multiple Gestalt features for various object recognition tasks. Many computer vision methods use features, which solve one specific problem or a class of problems. Gestalt features, being perceptually relevant and having direct relation to the geometric structure of the real world [27, 10], should be useful in most if not all object recognition problems[1]. Thus, even though the current work applies Gestalt features to one domain, the method can be potentially generalized to various object detection problems.

There seems to be even less research about the importance of using multiple partial gestalts[2] for better object detection. The proposed research addresses the problem by demonstrating experimentally that larger number of Gestalt features provides better large oil sand lump detection than smaller number of Gestalt features. Thus, the main contribution of the proposed method is in its results, which show a definite tendency for better detection performance when using a larger num-

[1] "not all geometric structures are perceptually relevant; a small list of relevant ones is given in Gestalt theory" ([6], p.3)

[2] "most salient objects or groups come to sight by several grouping laws ... The outcome of a partial gestalt detector is valid only when all other partial gestalts have been tested and the eventual conflicts dealt with" ([10], p.20)

ber of Gestalt features. The Gestalt theory predicted the results of my research[3]. Nevertheless, to my knowledge, the prediction was not confirmed experimentally until now.

The proposed method also addresses the fundamental problem of Gestalt collaboration by employing an automated machine learning framework to resolve Gestalt conflicts. More specifically, it uses decision tree classification to select acceptable detections. An important property of the decision tree method is that it does not require any parameters or thresholds to be set during the selection of the best candidate for an object that is being recognized/detected. The proposed method also uses motion within the Gestalt framework, which did not seem to be common among computer vision methods [21, 39].

The proposed method seemed to have good detection accuracy with large lump data that is a subclass of image depicting natural environments. The real-world images are, generally, very hard to use for object detection due to various problems: changing illumination, blurring due to camera shaking, meteorological conditions, etc. Also, using natural borders (edges) as the basic input of my algorithm preserves the natural shape of the objects unlike, for example, mathematical morphology methods.

7.2 Limitations

The proposed method has a number of limitations in both its design and experimental data. One of them is that the method requires *a priori* information about large lumps. If that information has substantial errors, then the whole method may have problems. For example, the sample of twenty training images may under-represent the real large lump characteristics (however, the risk was reduced by using random sampling), and this may heavily influence further the performance of the algorithm (for example, it might hide the differences between one- and two-feature output).

Also, the proposed method is limited to using only edge-based information.

[3] "objects that are conspicuous (*i.e.*, obvious to the eye or mind [19]) are very likely to be detected by several partial detectors (as predicted by Gestalt Theory), and a single detector does not give a definitive answer." ([4], p.12)

Using edges is well-justified, since such areas of high contrast are usually quite important for object detection. However, one could apply multiple Gestalt features to other basic input elements (*i.e.,* primitives), such as pixels, regions, and level lines that play an important role in object recognition.

The algorithm is computationally intensive, since it searches through all possible combinations of selected edges. It also is limited to at most three contours in one candidate shape and considers only seven widest edges. Another limitation is that the current implementation of the method can detect only one largest object.

7.3 Future Work

The proposed algorithm could undergo a number of optimizations and improvements. The method should be tested on additional various types of images to further support the claim about the importance of multiple Gestalt features. My work could be further extended by including more Gestalt features. The goal would be to compute all (or as many as possible) currently known partial gestalts (as per Cao [4]) and test them in other object recognition/detection tasks, in addition to large lump detection. For example, a partial gestalt of symmetry could be added, as defined by Kindratenko [20].

A better edge detector could substantially influence the final result of the proposed method, since edges are basic input elements to the algorithm. An edge detector created by Desolneux *et al.* [6] based on Gestalt principles seems to be a good alternative to Canny edge detector used in the current work. The edge detection method created by Cao [4] (which was based on the algorithm outlined by Desolneux *et al.*) could be even better alternative, since it seemed to favour good continuation contours that were not sensitive to smoothing. Therefore, Cao's method would work well for multiscale edge extraction (as a result, the proposed method could get rid of the scale parameter σ used in Canny edge detection). Obtained edges could be further simplified by fitting lines and arcs to them, as outlined by Rosin and West [37].

The selection of the best edges could also be improved. Current implementation does not do much edge filtering. Only edges appearing in two consecutive frames are selected for further processing and seven widest edges are set as the input for candidate shape creation. Therefore, a more intuitive process of edge selection could be devised. For example, edge orientation and concavity (for instance, using corresponding formulas from the shape theory described by [20]) could be taken into account, based on the assumption that most natural objects are convex.

Currently the proposed method computes Gestalt features from the edge information. Nevertheless, the idea of using multiple Gestalt features along with the machine learning decision module could also be applied to other primitives: pixels, regions, corners, alignments, texture, etc. These results could be used along with the results of the proposed method to increase the detection performance.

The overall speed of the method could be enhanced by moving the object detection into another module, which creates candidate shapes. In this case generation of all possible shapes could be avoided by stopping at the first acceptable candidate. The shape creation module itself could be improved by removing the limit on the number of edges used during candidate creation. The proposed method could also be expanded to allow for detection of multiple objects.

Bibliography

[1] M. Ali and D.A. Clausi. Using the canny edge detector for feature extraction and enhancement of remote sensing images. In *Proceedings of the International Geoscience and Remote Sensing Symposium (IGARSS)*, Sydney, Australia, 2001.

[2] M. B. Ben-Av and D. Sagi. Perceptual grouping by similarity and proximity: experimental results can be predicted by intensity autocorrelations. *Vision Res.*, 35:853–866, 1995.

[3] I. Bolakova. Pruning decision trees to reduce tree size. In *Traditional And Innovations In Sustainable Development Of Society*, pages 160–166, Rezekne, Latvia, 2002.

[4] F. Cao. Application of the Gestalt principles to the detection of good continuations and corners in image level lines. *Computing and Visualisation in Science*, 7:3–13, 2004.

[5] D. Chang, L. Dooley, and J.E. Tuovinen. Gestalt theory in visual screen design: a new look at an old subject. In *CRPITS '02: Proceedings of the Seventh world conference on computers in education conference on Computers in education: Australian topics*, pages 5–12, Darlinghurst, Australia, Australia, 2002. Australian Computer Society, Inc.

[6] A. Desolneux, L. Moisan, and J.-M. Morel. Edge detection by helmholtz principle. *Journal of Mathematical Imaging and Vision*, 14(3):271–284, 2001.

[7] A. Desolneux, L. Moisan, and J.-M. Morel. Partial gestalts. *CMLA*, 22, 2001.

[8] A. Desolneux, L. Moisan, and J.-M. Morel. Computational gestalts and perception thresholds. *Journal of Physiology-Paris*, 97(2):311–324, March 2003.

[9] A. Desolneux, L. Moisan, and J.-M. Morel. A grouping principle and four applications. *PAMI*, 25(4):508–513, 2003.

[10] A. Desolneux, L. Moisan, and J.-M. Morel. *Seeing, Thinking and Knowing*, chapter Gestalt Theory and Computer Vision, pages 71–101. A. Carsetti ed., Kluwer Academic Publishers, 2004.

[11] Wikipedia. The Free Encyclopedia. Ellipse. Website. www.wikipedia.org.

[12] C.G. Fowler, R. Kube, and T. Kamm. Progress report: crusher production limitations due to frozen lump jams. Technical Report 29(11), Syncrude Research Department, Edmonton, Alberta, Canada, 2000.

[13] I. Galkin, B. Reinisch, G. Grinstein, G. Khmyrov, A. Kozlov, X. Huang, and Sh. Fung. Automated exploration of the radio plasma imager data. *Journal of Geophysical Research*, 109:A12210+, December 2004.

[14] E. B. Goldstein. *Sensation and Perception*. Brooks/Cole Publ., Pacific Grove, CA, fifth edition, 1999.

[15] R.C. Gonzalez and R.E. Woods. *Digital Image Processing*. Addison-Wesley Publishing Company, USA, 1992.

[16] M. Heath, S. Sarkar, T. Sanocki, and K. Bowyer. Comparison of edge detectors: A methodology and initial study. In *Proceedings of the 1996 Conference on Computer Vision and Pattern Recognition (CVPR '96)*, pages 143–148, Washington, DC, USA, 1996. IEEE Computer Society.

[17] H. Helson. The fundamental propositions of gestalt psychology. *Psychological Review*, 40:13–32, 1933.

[18] J. Huart and P. Bertolino. Similarity-based and perception-based image segmentation. In *International Conference on Image Processing (ICIP) 2005*, pages 1148–1151, 2006.

[19] Merriam-Webster Inc. Merriam-webster online. Website. http://www.m-w.com.

[20] V. Kindratenko. *Development and Application of Image Analysis Techniques for Identification and Classification of Microscopic Particles*. PhD thesis, University of Antwerpen, Antwerpen, Belgium, 1997.

[21] K. Korimilli and S. Sarkar. Motion segmentation based on perceptual organization of spatio-temporal volumes. In *ICPR*, pages 3852–3857, 2000.

[22] S.-W. Lee, J.G. Choi, and S.-D. Kim. Scene segmentation using a combined criterion of motion and intensity. *Optical Engineering*, 36(8):2346–2352, 1997.

[23] S. Lehar. *The World in Your Head: A Gestalt View of the Mechanism of Conscious Experience*. Lawrence Erlbaum Associates, August 2002.

[24] A. Litvin and W.C. Karl. Image segmentation based on prior probabilistic shapes models. In *Proceedings of 2002 IEEE International Conference on Acoustic Speech and Signal Processing (ICASSP)*, Orlando, 2002.

[25] Y.-F. Ma and H.-J. Zhang. Contrast-based image attention analysis by using fuzzy growing. In *MULTIMEDIA '03: Proceedings of the eleventh ACM international conference on Multimedia*, pages 374–381, New York, NY, USA, 2003. ACM Press.

[26] D. Marr. Vision: A computational investigation into the human representation and processing of visual information. In *W.H. Freeman*, 1982.

[27] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. Technical report, EECS Department, University of California, Berkeley, 2001.

[28] A. Maßmann, S. Posch, G. Sagerer, and D. Schlüter. Using markov random fields for contour-based grouping. In *ICIP*, volume II, pages 207–210. IEEE, 1997.

[29] Mathworks. Matlab. Website. http://www.mathworks.com/.

[30] D.P. Mukherjee, Y. Potapovich, I. Levner, and H. Zhang. Ore image segmentation by learning image and shape features. *Unpublished*, 2006.

[31] T.W. Nattkemper, H. Wersing, W. Schubert, and H. Ritter. Fluorescence micrograph segmentation by gestalt-based feature binding. In *Proc. of the Int. Joint Conf. on Neur. Netw. (IJCNN)*, volume 1, pages 248–254, Como, Italy, 2000.

[32] H. Raiffa and R. Schlaifer. *Applied Statistical Decision Theory*. Harvard Business School, Cambridge, MA, 1961.

[33] X. Ren and J. Malik. Learning a classification model for segmentation. In *ICCV*, pages 10–17, 2003.

[34] M. Reybrouck. Gestalt concepts and music: Limitations and possibilities. In *Music, Gestalt, and Computing - Studies in Cognitive and Systematic Musicology*, pages 57–69, London, UK, 1997. Springer-Verlag.

[35] G. Richards. *Putting psychology in its place: A critical historical overview*. Taylor and Francis, London: Routledge, second edition, 2002.

[36] I. Rock and S. Palmer. The legacy of gestalt psychology. *Scientific American*, December:84–90, 1990.

[37] P.L. Rosin and G.A.W. West. Nonparametric segmentation of curves into various representations. *IEEE Trans. Pattern Anal. Mach. Intell.*, 17(12):1140–1153, 1995.

[38] S. Sarkar. Learning to form large groups of salient image features. In *CVPR*, pages 780–786, 1998.

[39] S. Sarkar, D. Majchrzak, and K. Korimilli. Perceptual organization based computational model for robust segmentation of moving objects. *Comput. Vis. Image Underst.*, 86(3):141–170, 2002.

[40] N. Sebe and M.S. Lew. *Robust Computer Vision Theory and Applications*. Kluwer, April 2003.

[41] H.A. Simon. The information processing explanation of Gestalt phenomena. *Computers in Human Behavior*, 2:241–255, 1986.

[42] ImageGrab 3.0 Software. Website. http://paul.glagla.free.fr/imagegrab_en.htm.

[43] WinDV Software. Website. http://windv.mourek.cz.

[44] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis, and Machine Vision*. International Thomson Computer Press, London, 1996.

[45] M. Tabb and N. Ahuja. Multiscale image segmentation by integrated edge and region detection. *IEEE Transactions on Image Processing*, 6(5):642–655, 1997.

[46] S.P. Teeuwsen, I. Erlich, and M.A. El-Sharkawi. Decision tree based oscillatory stability assessment for large interconnected power systems. In *PSCE*, volume II, pages 1089–1094. IEEE, 2004.

[47] B.M. Thorne and T.B. Henley. *Connections in the History and Systems of Psychology*. Houghton Mifflin, Boston, second edition, 2001.

[48] S. Ullman. *The Interpretation of Visual Motion*. MIT Press, 1979.

[49] W. Viney and D.B. King. *A History of psychology: Ideas and context*. Allyn and Bacon, Boston, third edition, 2003.

[50] A.B. Watson and A.J. Ahumada. Model of human visual-motion sensing. *J. Opt. Soc. Am. A*, 2(2):322–342, 1985.

[51] M. Wertheimer. A gestalt perspective on computer simulations of cognitive processes. *Computers in Human Behavior*, 1:19–33, 1985.

[52] S.C. Zhu. Embedding gestalt laws in markov random fields. *PAMI*, 21(11):1170–1187, November 1999.

[53] N. Zlatoff, B. Tellez, and A. Baskurt. Pre-Attentive Vision: Combining Gestalt Laws with the Notion of Belief. Technical report, LIRIS UMR 5205 CNRS/INSA de Lyon/Universit Claude Bernard Lyon 1/Universit Lumire Lyon 2/Ecole Centrale de Lyon, November 2004.

Appendix A

Additional Results

Figures A.1 and A.2 contain mean (from three different decision tree splitting criteria) results for each Gestalt feature combination used in the experiments: intensity ("i"), motion ("m"), shape ("s"), intensity and motion ("i+m"), intensity and shape ("i+s"), motion and shape ("m+s"), intensity, motion and shape ("i+m+s"). Figures A.3-A.8 contain results displayed separately for each of three decision tree splitting criteria: Twoing, maximum deviance reduction and Gini's diversity index. Tables A.1, A.2, and A.3 contain numeric results obtained using Twoing, maximum deviance reduction and Gini's diversity index splitting criteria. Figures A.9, A.10, A.11, and A.12 contain some visual detection results.

Gestalt features	Accuracy	Precision	Recall	False positive rate
i	0.60133	0.36716	0.44537	0.33134
m	0.63928	0.38087	0.31397	0.2203
s	0.65304	0.41577	0.37206	0.22567
i+m	0.62636	0.38924	0.42047	0.28478
i+s	0.67848	0.45161	0.30982	0.16239
m+s	0.65513	0.42097	0.38313	0.22746
i+m+s	0.68307	0.46794	0.37344	0.18328

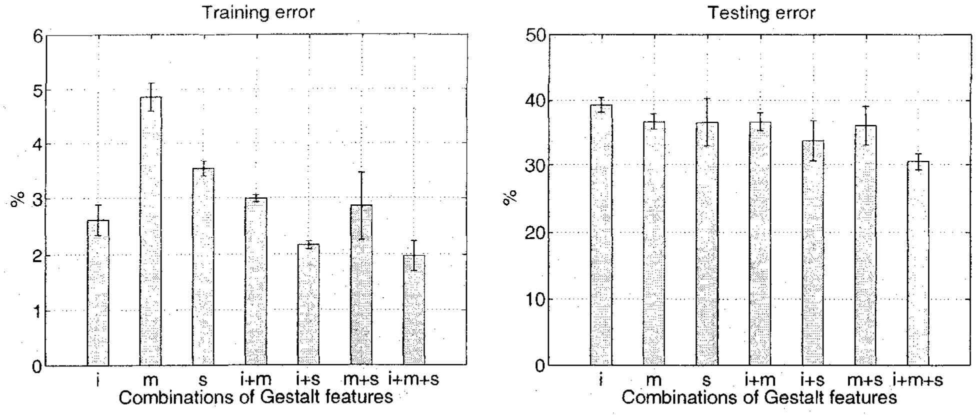
Table A.1: Results obtained using Twoing decision tree rule.

Gestalt features	Accuracy	Precision	Recall	False positive rate
i	0.61927	0.38443	0.43707	0.30209
m	0.62093	0.36443	0.34578	0.2603
s	0.59633	0.38118	0.54357	0.3809
i+m	0.64762	0.38745	0.29046	0.19821
i+s	0.63011	0.38889	0.39696	0.26925
m+s	0.60926	0.37868	0.46196	0.32716
i+m+s	0.696	0.49414	0.34993	0.15463

Table A.2: Results obtained using a maximum deviance reduction rule.

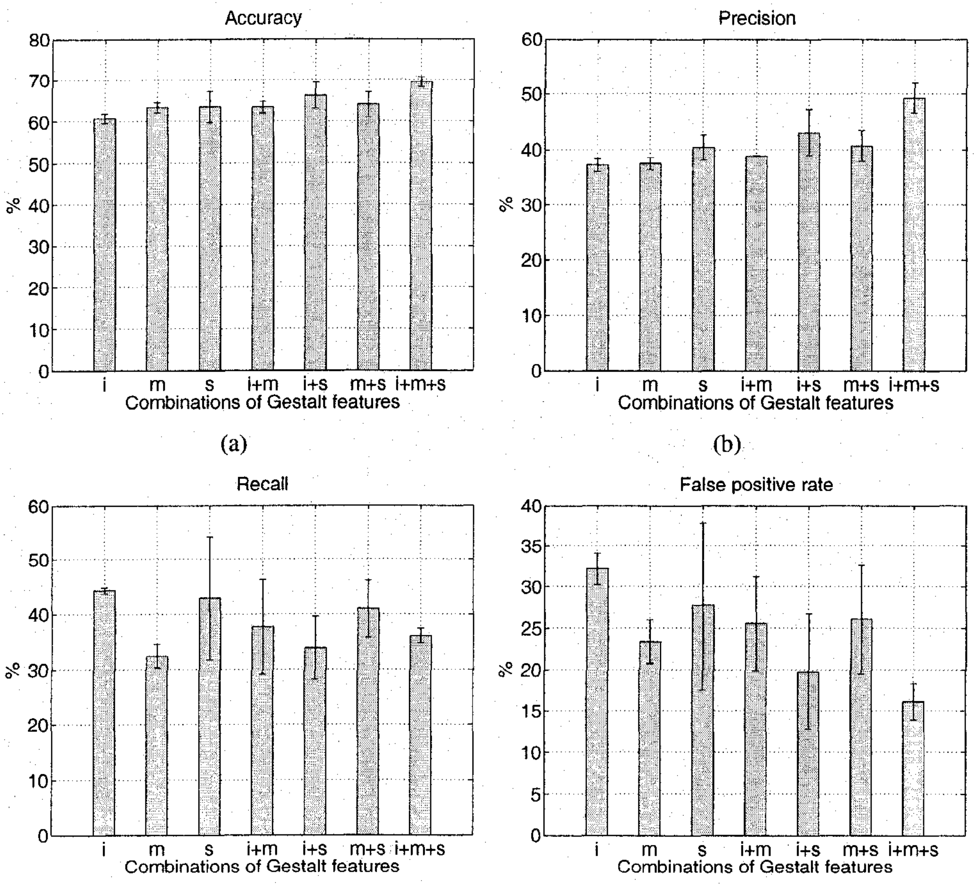
Gestalt features	Accuracy	Precision	Recall	False positive rate
i	0.60133	0.36716	0.44537	0.33134
m	0.63928	0.38087	0.31397	0.2203
s	0.65304	0.41577	0.37206	0.22567
i+m	0.62636	0.38924	0.42047	0.28478
i+s	0.67848	0.45161	0.30982	0.16239
m+s	0.65513	0.42097	0.38313	0.22746
i+m+s	0.70475	0.51485	0.35961	0.14627

Table A.3: Results obtained using Gini's diversity index.



(a) (b)

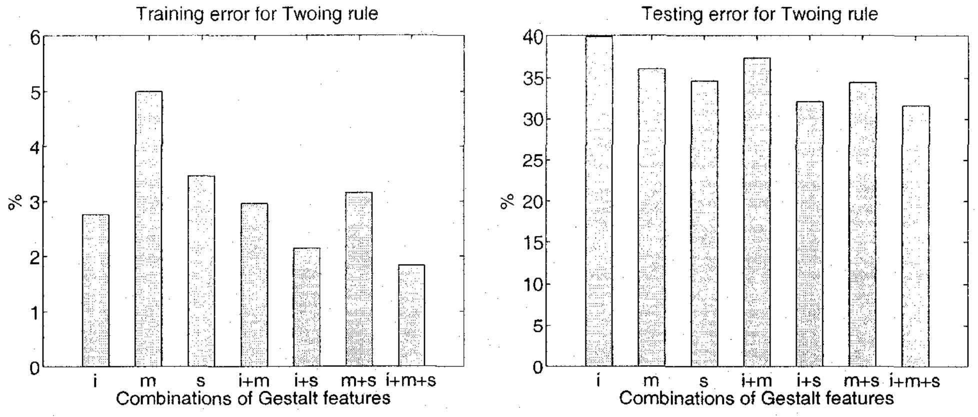
Figure A.1: Mean training and testing error for each combination of Gestalt features.



(a) (b)

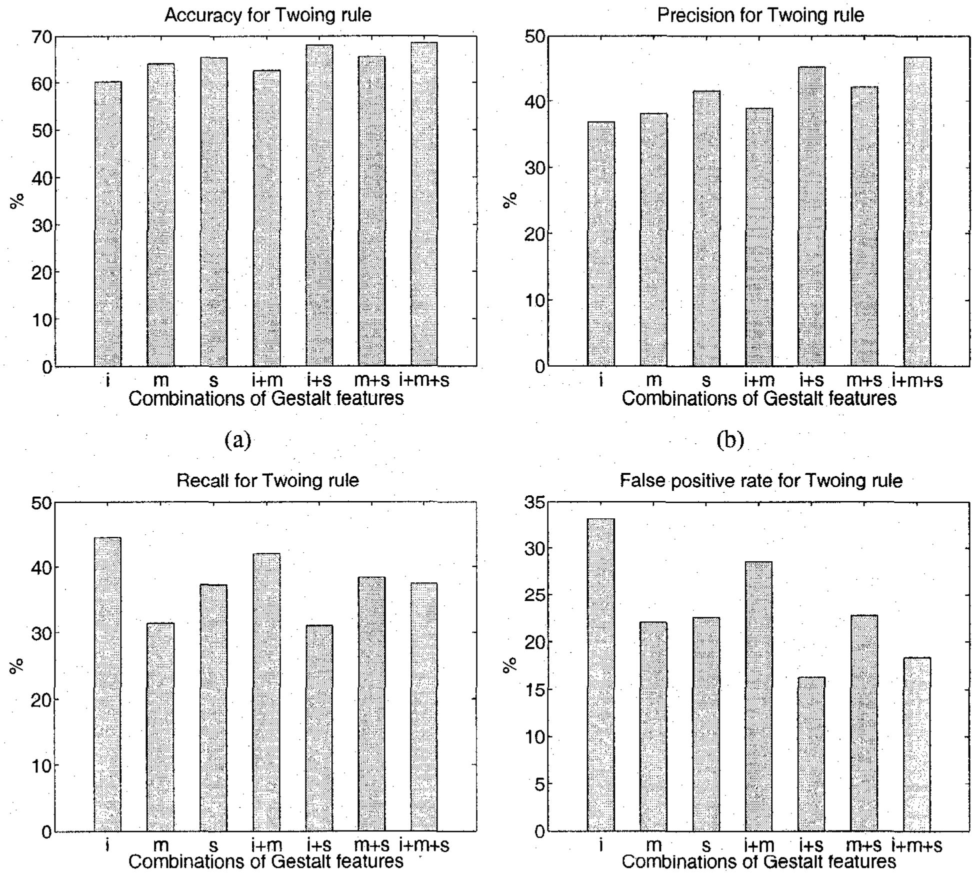
(c) (d)

Figure A.2: Mean results for each combination of Gestalt features.



(a) (b)

Figure A.3: Training and testing error obtained using Twoing rule as a decision tree splitting criterion.



(a) (b)

(c) (d)

Figure A.4: Results obtained using Twoing rule as a decision tree splitting criterion.

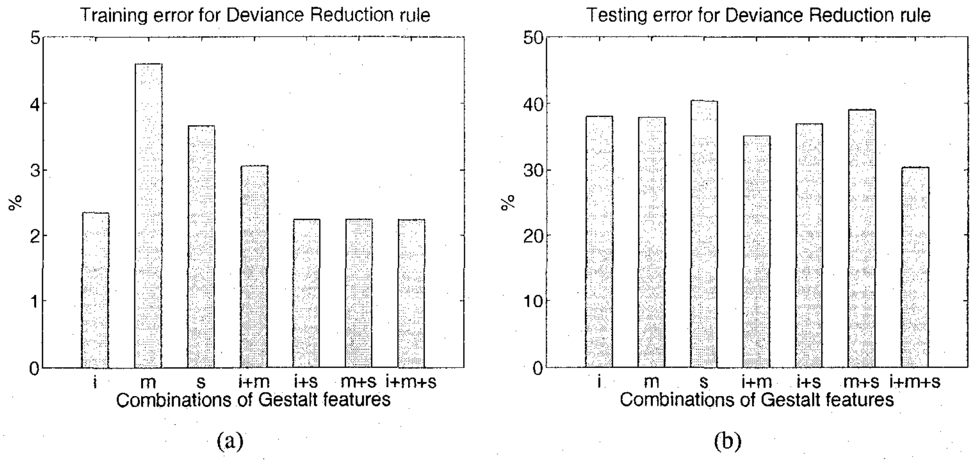


Figure A.5: Training and testing error obtained using a maximum deviance reduction rule as a decision tree splitting criterion.

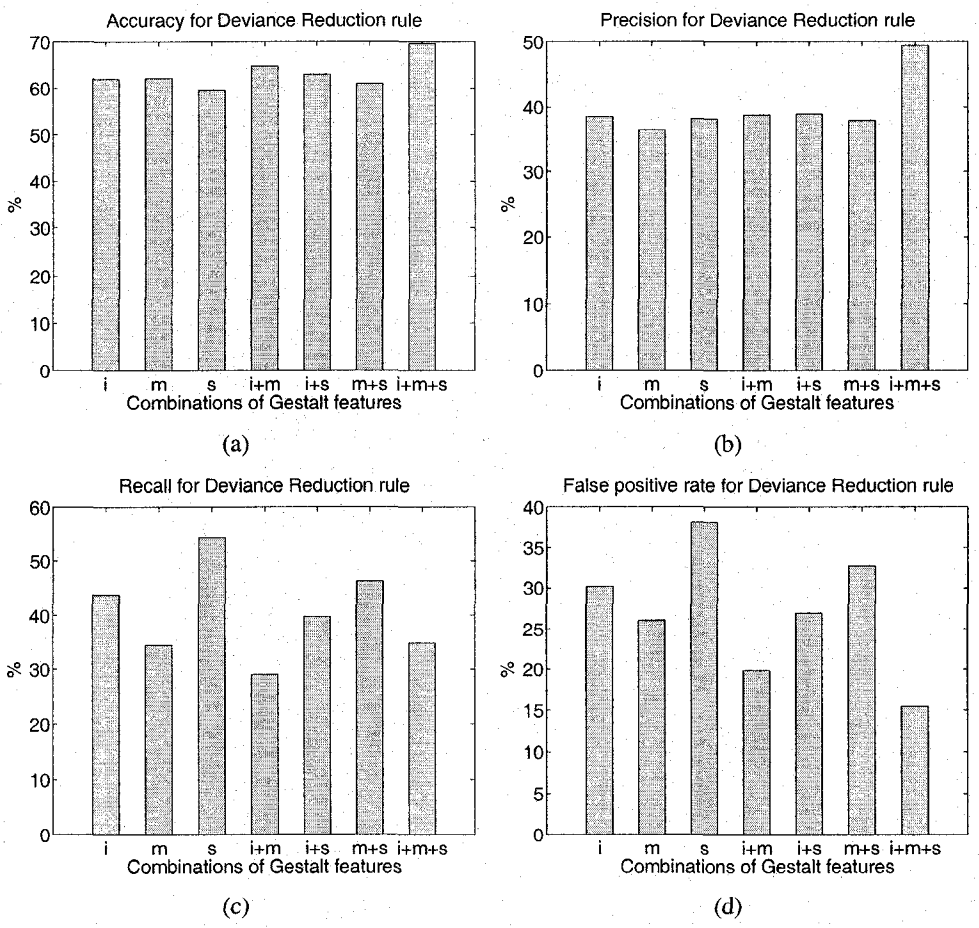


Figure A.6: Results obtained using a maximum deviance reduction rule as a decision tree splitting criterion.

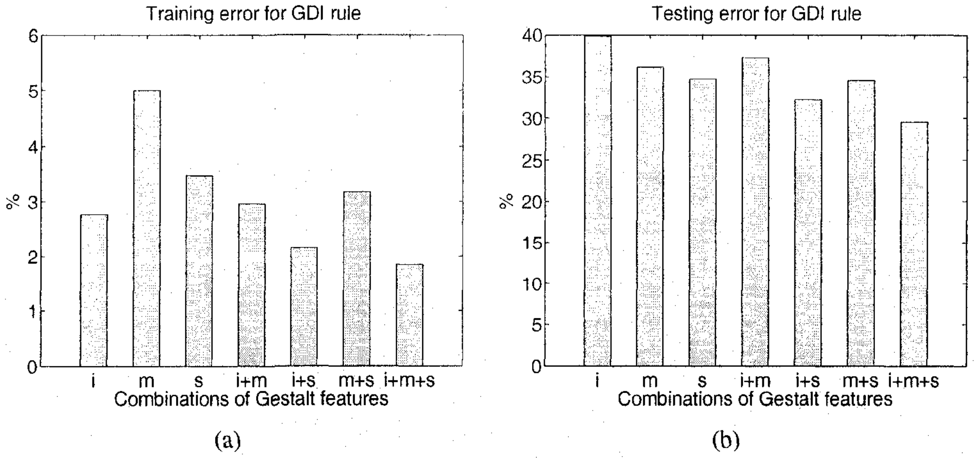

(a) (b)

Figure A.7: Training and testing error obtained using Gini's diversity index as a decision tree splitting criterion.

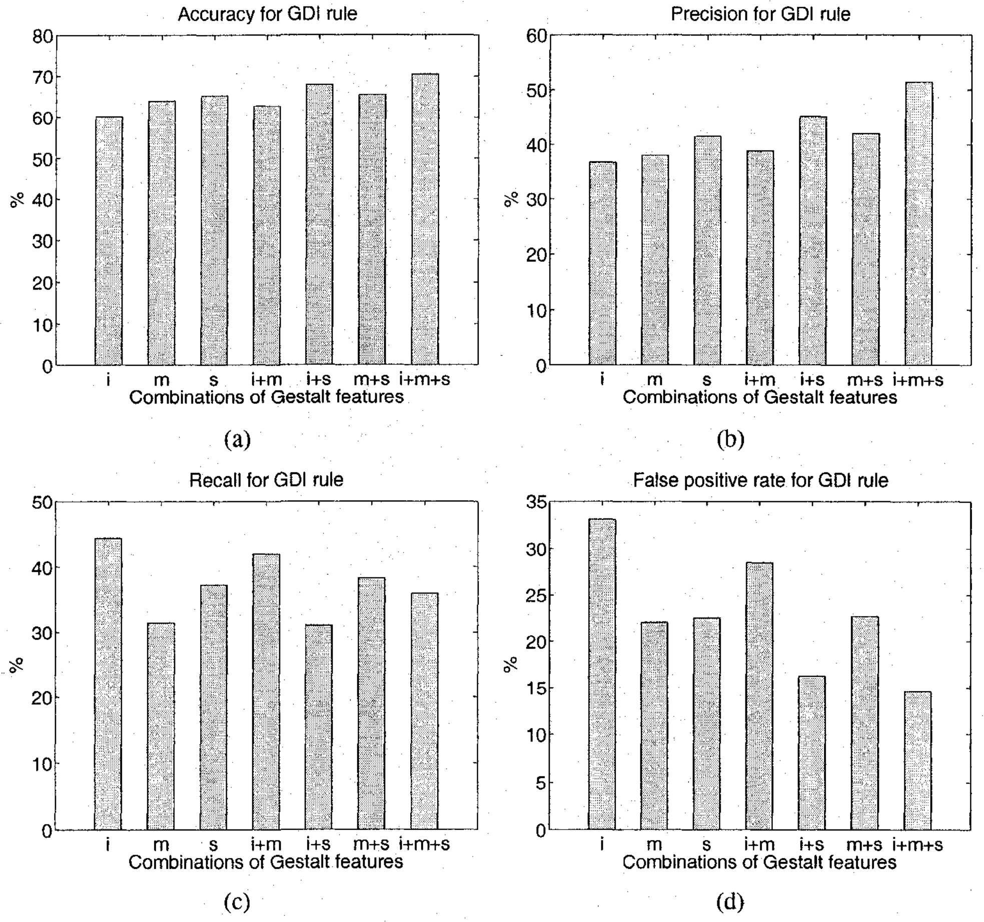

(a) (b) (c) (d)

Figure A.8: Results obtained using Gini's diversity index as a decision tree splitting criterion.

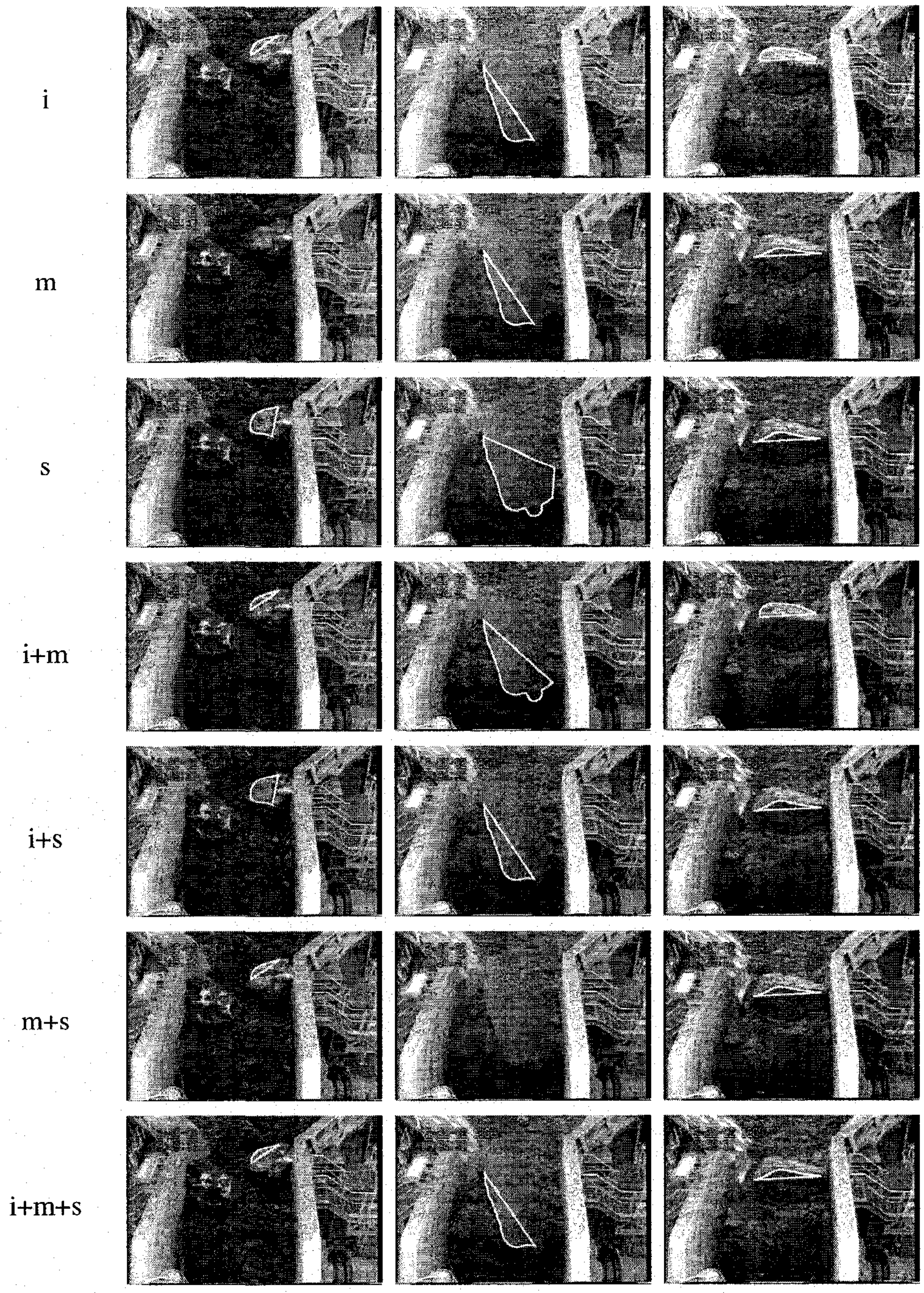


Figure A.9: Detection results of some training sample images that contain large lumps. The results were obtained using Twoing decision rule. Each row contains results for one configuration of Gestalt features from the following list: intensity, motion, shape, intensity+motion, intensity+shape, motion+shape, intensity+motion+shape.

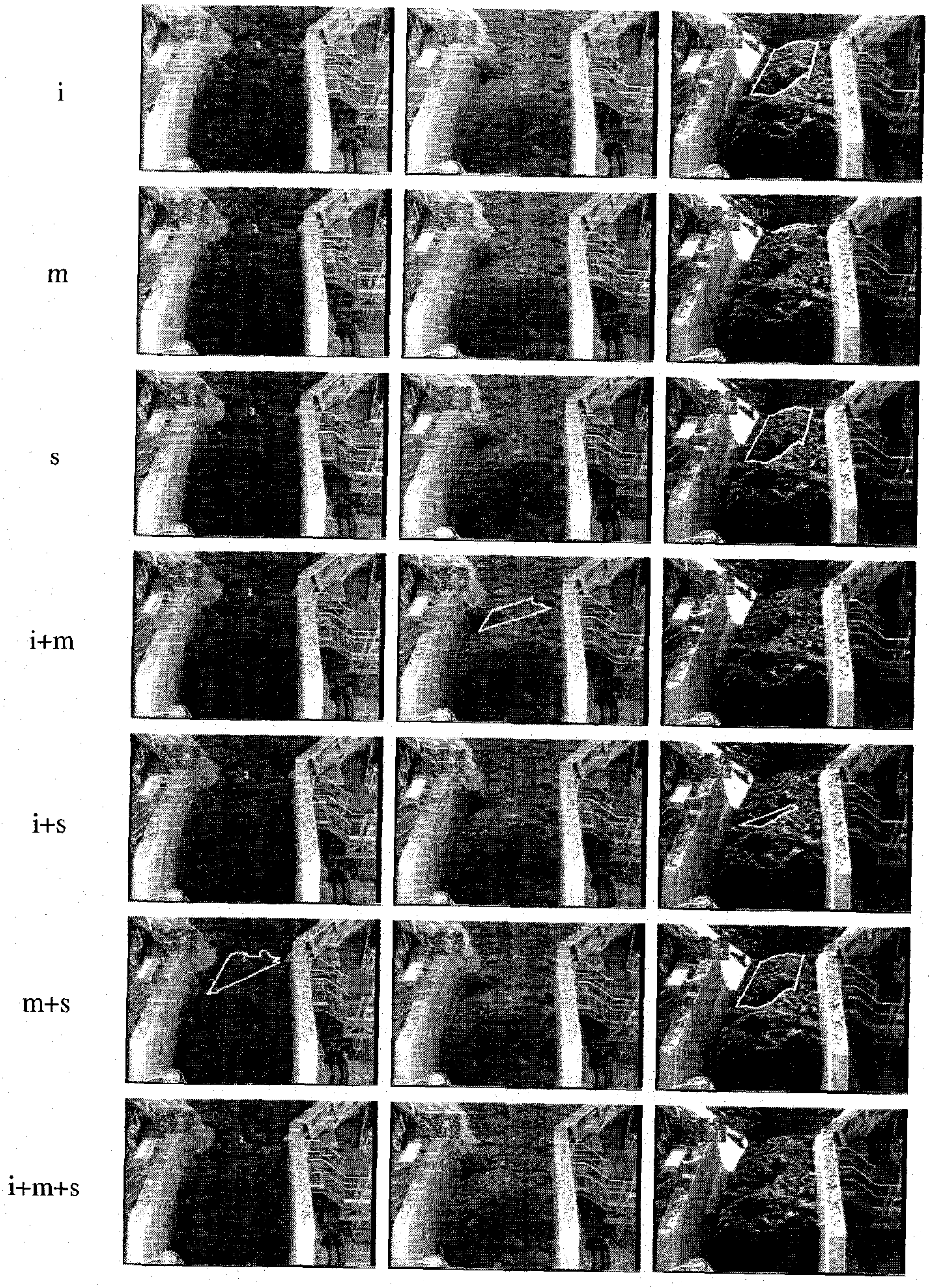


Figure A.10: Detection results of some training sample images that do not contain large lumps. The results were obtained using Twoing decision rule. Each row contains results for one configuration of Gestalt features from the following list: intensity, motion, shape, intensity+motion, intensity+shape, motion+shape, intensity+motion+shape.

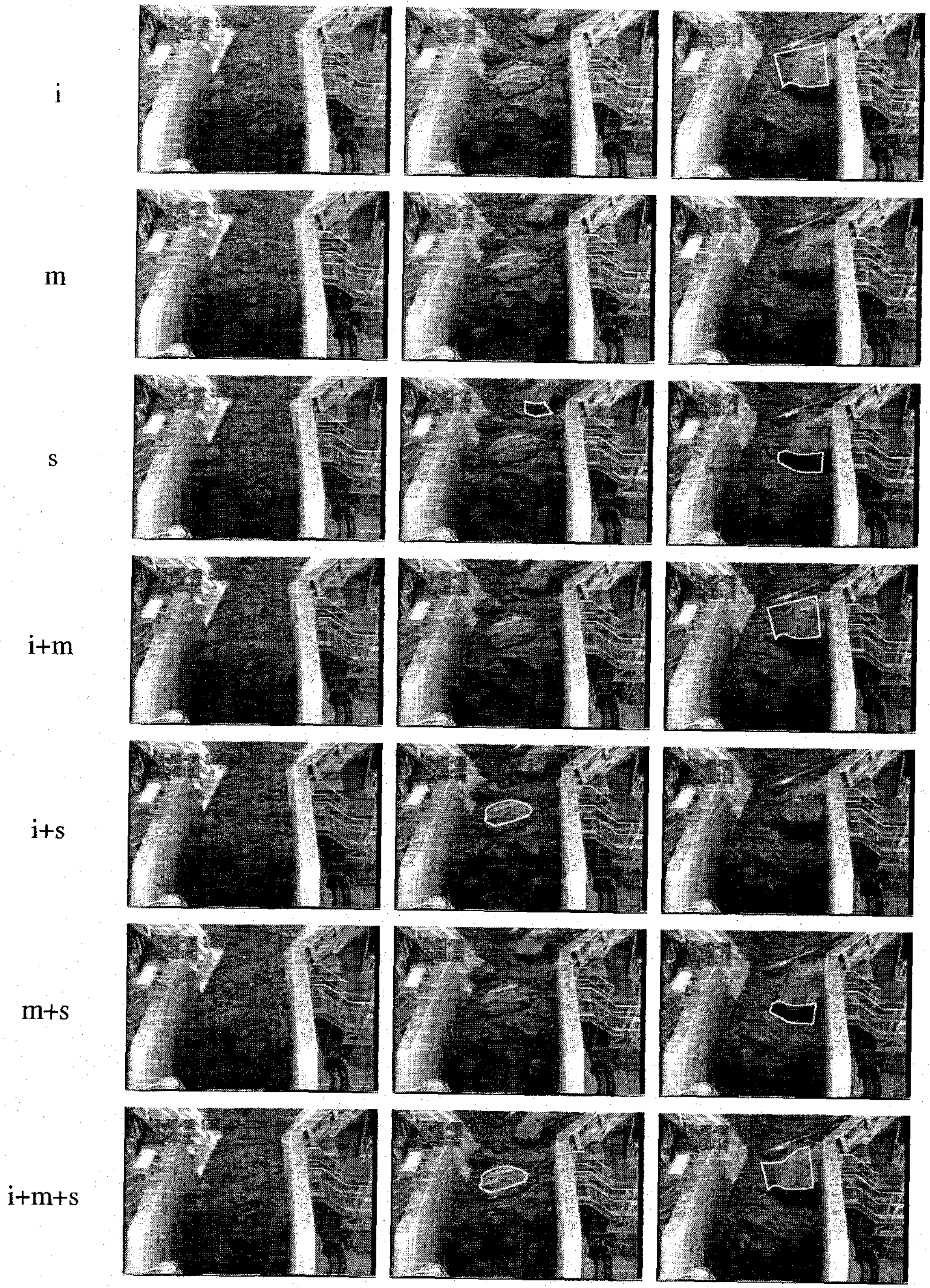


Figure A.11: Detection results of randomly selected images 291, 1073, 1093. The results were obtained using Twoing decision rule. Each row contains results for one configuration of Gestalt features from the following list: intensity, motion, shape, intensity+motion, intensity+shape, motion+shape, intensity+motion+shape.

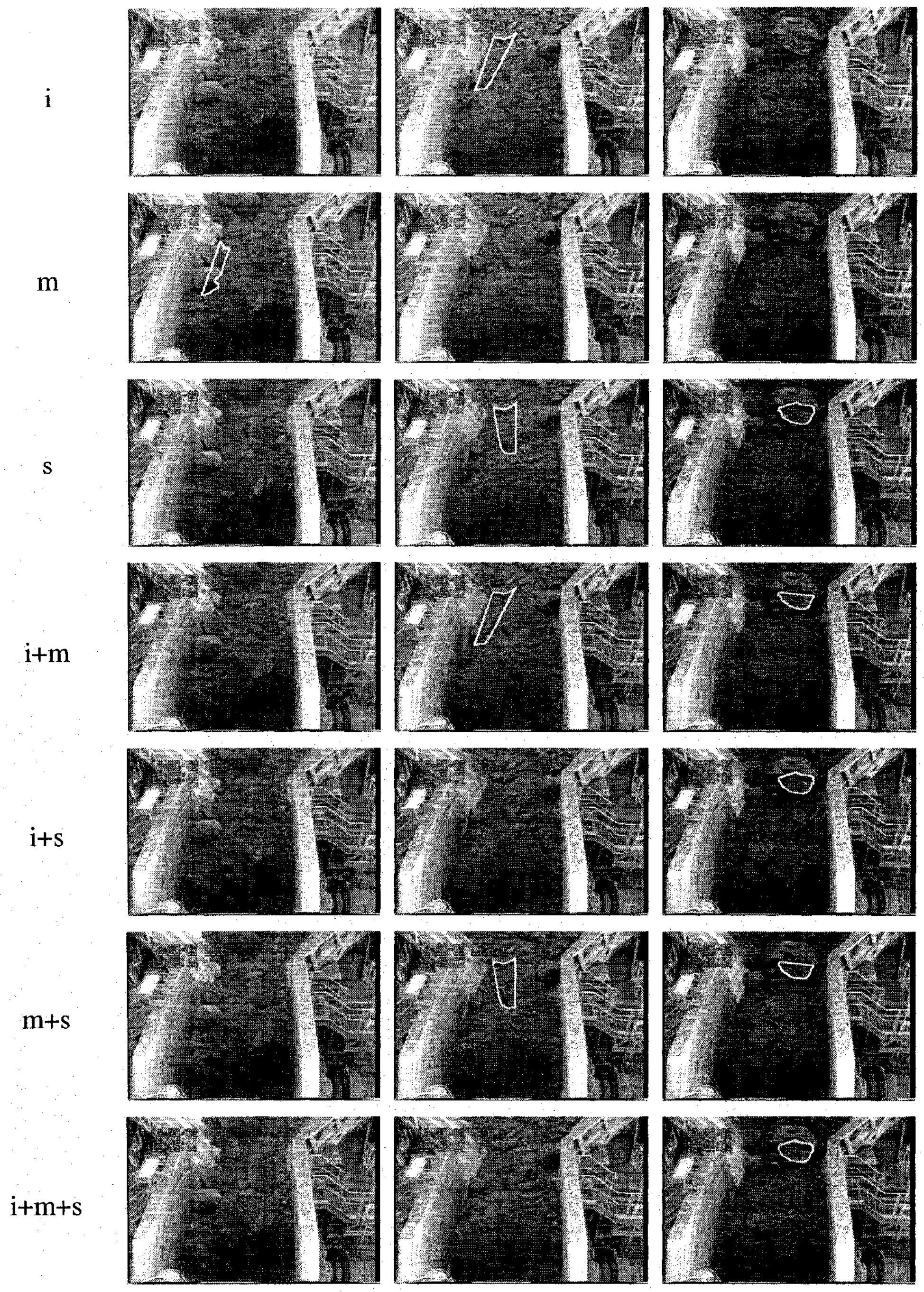


Figure A.12: Detection results of randomly selected images 1219, 1735, 2356. The results were obtained using Twoing decision rule. Each row contains results for one configuration of Gestalt features from the following list: intensity, motion, shape, intensity+motion, intensity+shape, motion+shape, intensity+motion+shape.