PROCEEDINGS

SPIE—The International Society for Optical Engineering

Advances in Image Compression and Automatic Target Recognition

Andrew G. Tescher
Chair/Editor

30–31 March 1989
Orlando, Florida

SPIE P PROCEEDINGS SERIES

Volume 1099

PROCEEDINGS
SPIE—The International Society for Optical Engineering

Advances in Image Compression and Automatic Target Recognition

Andrew G. Tescher
Chair/Editor

30–31 March 1989
Orlando, Florida

Sponsored by
SPIE—The International Society for Optical Engineering

Cooperating Organizations
Applied Optics Laboratory/New Mexico State University
Center for Applied Optics Studies/Rose-Hulman Institute of Technology
Center for Applied Optics/University of Alabama in Huntsville
Center for Electro-Optics/University of Dayton
Center for Excellence in Optical Data Processing/Carnegie Mellon University
Center for Microwave-Lightwave Engineering/Drexel University
Center for Research in Electro-Optics and Lasers/University of Central Florida
Jet Propulsion Laboratory/California Institute of Technology
Optical Sciences Center/University of Arizona
Optoelectronic Computing Systems Center/University of Colorado,
Colorado State University

Published by
SPIE—The International Society for Optical Engineering
P.O. Box 10, Bellingham, Washington 98227-0010 USA
Telephone 206/676-3290 (Pacific Time) • Telex 46-7053

SPIE PROCEEDINGS SERIES

Volume 1099

SPIE (The Society of Photo-Optical Instrumentation Engineers) is a nonprofit society dedicated to advancing engineering and scientific applications of optical, electro-optical, and optoelectronic instrumentation, systems, and technology.

The papers appearing in this book comprise the proceedings of the meeting mentioned on the cover and title page. They reflect the authors' opinions and are published as presented and without change, in the interests of timely dissemination. Their inclusion in this publication does not necessarily constitute endorsement by the editors or by SPIE.

Please use the following format to cite material from this book:
Author(s), "Title of Paper," *Advances in Image Compression and Automatic Target Recognition,* Andrew G. Tescher, Editor, Proc. SPIE 1099, page numbers (1989).

Library of Congress Catalog Card No. 89-61610
ISBN 0-8194-0135-8

Address inquiries and notices to Director of Publications, SPIE, P.O. Box 10, Bellingham, WA 98227-0010 USA.

ADVANCES IN IMAGE COMPRESSION
AND AUTOMATIC TARGET RECOGNITION

Volume 1099

CONFERENCE COMMITTEE

Chair
Andrew G. Tescher, Lockheed Palo Alto Research Laboratories

Cochairs
Richard L. Baker, University of California/Los Angeles
Mohan M. Trivedi, University of Tennessee/Knoxville

Session Chairs
Session 1—Target Recognition Systems
Andrew G. Tescher, Lockheed Palo Alto Research Laboratories

Session 2—Target Recognition Concepts
Mohan M. Trivedi, University of Tennessee/Knoxville

Session 3—Image Compression
Richard L. Baker, University of California/Los Angeles

Session 4—Vision and Pattern Recognition
Richard S. Loe, Lockheed Missiles & Space Company

Conference 1099, *Advances in Image Compression and Automatic Target Recognition*, was part of a six-conference program on Signal and Image Processing held at SPIE's 1989 Technical Symposia on Aerospace Sensing. The other conferences were

Conference 1096, *Digital Signal Processing, Association, and Tracking of Point Source, Very Small, and Cluster Targets*
Conference 1098, *Aerospace Pattern Recognition*
Conference 1100, *Sensor Fusion II*
Conference 1102, *Optical Technology for Microwave Applications IV*
Conference 1105, *Materials for Optical Switches, Isolators, and Limiters.*

Program Chair: **Louis C. Marquet,** Atlantic Aerospace Electronics Corporation

ADVANCES IN IMAGE COMPRESSION
AND AUTOMATIC TARGET RECOGNITION

Volume 1099

INTRODUCTION

The fields of image compression and target recognition have evolved individually into important technologies. It is less appreciated, however, that many of the processing tools are common to these two important disciplines. The identification of important image features is fundamental to an efficient target recognition system as well as to a practical adaptive image compression procedure.

The SPIE 1989 Conference on Advances in Image Compression and Automatic Target Recognition was a novel attempt to bring together researchers in these two important fields for the purpose of exposing them to the potentially greater available resources outside their own specializations.

The technical program covered a good cross section of the relevant technologies.The automatic target recognition activities are covered in two sessions. One session addressed image compression and the remaining program covered vision and pattern recognition issues.

The efforts of the program organizers, session chairs, and the contributors are sincerely appreciated.

Andrew G. Tescher
Lockheed Palo Alto Research Laboratories

ADVANCES IN IMAGE COMPRESSION
AND AUTOMATIC TARGET RECOGNITION

Volume 1099

CONTENTS

(continued)

ADVANCES IN IMAGE COMPRESSION
AND AUTOMATIC TARGET RECOGNITION

Volume 1099

SESSION 1

Target Recognition Systems

Chair
Andrew G. Tescher
Lockheed Palo Alto Research Laboratories

Recognition of Cars on Color Aerial Images for traffic Analysis.

J.T.Van Gael*, P.Suetens†, F.Fierens, P. Wambacq†, A.Oosterlinck.
ESAT-MI2, Kardinaal Mercierlaan 94, B-3030 Heverlee, Belgium

February 10, 1989

ABSTRACT

Due to the high availability of aerial images, the automated interpretation has become an increasing need. One of the application fields is the interpretation of aerial images of cities. The interpretation allows for a better understanding of the traffic patterns. We have chosen this application because its a challenging task involving complex images. This paper describes the need for techniques beyond the pure numerical and statistical methods. It describes how the use of symbolic oriented rules combined with humanlike reasoning mechanisms can drastically improve results.

1. Problem statement.

1.1. Introduction.

The analysis or interpretation of aerial and satellite images has gained great interest [3] and has become a major research topic in computer vision. The main reason for this, is the extreme high availability of these images and the need to analyze them. Their complexity makes them also an excellent testbed for advanced research in computer vision. One of the applications in aerial image interpretation involves the study of traffic patterns in cities. Due to the increasing traffic density and the inevitable traffic congestions, people responsible for the organization of traffic flow need to know the existing traffic patterns in order to be able to take actions. One of the best ways to do this is by analyzing aerial images.

1.2. The Data.

Photographs are taken from an airborne camera at an altitude of approximately 600 m. Two images are taken with an interval of approximately 3 seconds. These photographs are then digitized with an approximate resolution of 3 pixels per meter. The digitization process produces three spectral bands. It was possible to perform a rough geometrical correction on both images such that image locations between both images corresponded roughly. The images are from downtown Brussels at rush hour.

1.3. Goal of the system.

As described above, the final aim is to obtain an idea of the traffic patterns. In order to do this, one needs to know where there are cars and how fast they drive. This information is at the basis for all relevant aspects of traffic analysis. The system that will be described will have as output, a list of cars and their respective locations and speed. The basic idea is that the system will try to find all cars in both images and see which car in one image corresponds to which car in the other image. The difference in location and the difference in time between the two images, allows us to derive the speed of all cars.

---

*J.Van Gael is also associated to the Royal Military Academy, Brussels

†P.Suetens and P.Wambacq are also research associates of the National Fund for scientific Research. Belgium

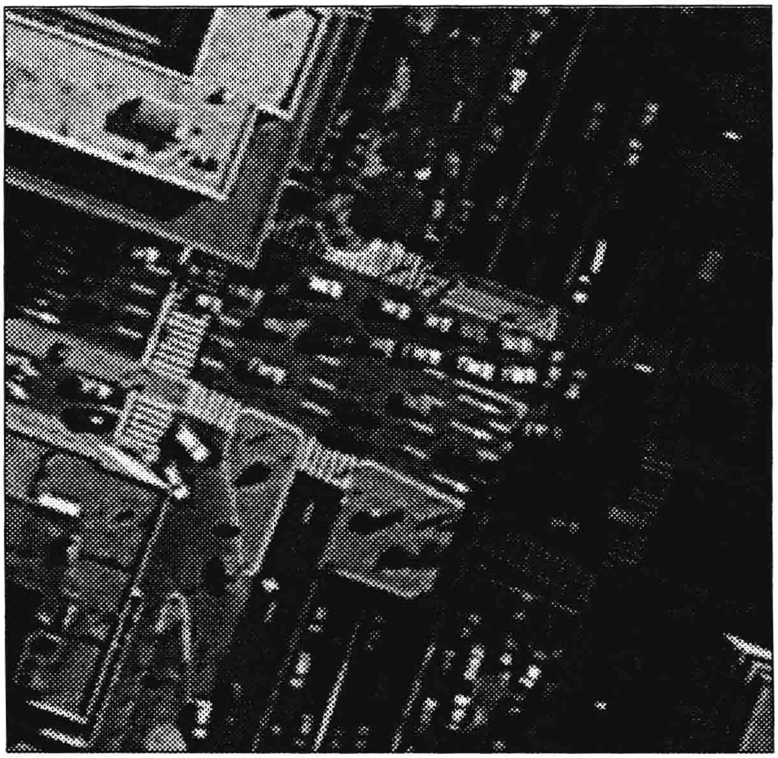

Figure 1: *Part (256 by 256 pixels) of an aerial image (red band)*

2. System structure.

2.1. Feature based classification.

2.1.1. Segmentation We first have tried to apply existing computer vision techniques in order to solve the problem. The first step towards interpretation lies in a good segmentation. We have therefore done a study as to which segmentation techniques were suitable for our application. The Ohlander region finding algorithm [4] turned out to perform very well on these images, compared to other region finding techniques. This is of course not to mean that it wouldn't be possible to enhance these segmentation results. The general idea was to make an abstraction of the pixel level to a level where we have reasonably meaningful regions.

2.1.2. Feature extraction. The Ohlander algorithm is a recursive region splitting algorithm that takes as input a number of spectral bands and has as output one labeled image where each pixel belongs to exactly one region. Such a region shows some stable pixel value properties over the complete number of input images. After we know the label image, we can derive some characteristic values for each region. Among other characteristics, we used :

- Location (Center of gravity).
- Size (In pixels).
- Mean value in the R,G and B band.
- Maximum values in these bands.
- Minimum bounding rectangle. (Description of a possibly tilted rectangle which has as property that among all possible rectangles it covers best the considered region. The size of this rectangle and the real size of the region gives us a very good indication on how well a certain region fits a rectangle. This is of course a very useful cue in the detection of cars.

2.1.3. Feature space and detection. After having determined these regions and their corresponding features, it seems obvious to try to detect cars based on these features. A car is a meaningful object that shows some photometric differences with its surroundings and hence can be expected to be delineated by a single region. A region can be characterized in an n-dimensional feature space. Each dimension represents a certain feature (but not the location of course). Interpreting the image could then be done by determining the boundaries of all the subspaces, such that each subspace represents one class of meaningful objects. The determination of these boundaries can be done empirically or statistically. We have determined the boundaries empirically and came at first sight to rather encouraging results. We were able to detect about 80 % of all cars.

2.1.4. Problems with feature based classification. Unfortunately the number of false alarms was unacceptably high, in the order of 50 %. The hope was that the subspace, part of our feature space, which embraces all the points representing cars, would have no intersection with those subspaces that embraced other meaningful objects such as e.g. roadmarks. We have demonstrated however that it was impossible for our data to obtain convex non intersecting subspaces. In order to determine the minimal subspace containing all the cars, we would have to allow for non convex subspaces. Not only would the decision criteria become awkward, but the boundaries would become so ad hoc that this decision criteria would be all but general and hence useless. It became clear that it would never be possible to come to a satisfactory solution based on only feature based inspection.

2.2. A contextual reasoning based system.

2.2.1. The need for contextual reasoning. Our unsatisfying results, based only on the features of individual regions will not come as a surprise to researchers involved in computer vision who know that segmentation is (almost?) never perfect. This is clearly expressed in [2] where the signal-symbol paradigm is explained. The question becomes how to interpret regions and hence how to detect cars if it is not possible to do this based on the features of individual regions. A careful analysis of the interpretation process shows that trying to interpret the images based on feature inspection, was comparable to the interpretation of the images by a human expert who would look at the images with a magnifying-glass. The person would very clearly see the details of a particular spot in the image but would not see the *whole picture*. It is obvious that everybody would have a hard time interpreting an image that way. Probably we would be unable to see the difference in some cases between a big roadmark and a car of the same color as the roadmark. To overcome this difficulty we must step back andf look at the *whole picture*. We will give a meaning to a part of the image in terms of how it relates to other objects in the image. If a person asked why he decides that something is a car, the answer could be something of the kind:

```
That must be a car because
that blob is on the road,
it has not the same color as the road itself,
I have already detected cars in front of and behind that blob
and I don't see any roadmarks on that specific lane.
```

2.2.2. General approach. We to start from the segmentation results, even if we know that the segmentation results can not be perfect. We will therefore take a conservative attitude in that we will only decide for a limited number of regions if they can be classified as cars (or as an other class such as roadmarks) according to some very restrictive feature based decision criteria. In our feature space this is translated as determining a small subspace for each class of objects. This also implies that the space between these subspaces is characterized as an area of uncertainty. Most regions will have features such that they fall in this uncertain area. To be able to interpret these other uncertain regions, we have implemented a system that was capable of applying rules, such as the one described above, in an existing context. To be able to start this reasoning process, our first context will be based on the partial results of the feature based interpretations of a limited number of regions. The system will then gradually apply these rules and we will get each time a more complete context. The final context will be the complete interpretation of the two images.

2.2.3. Knowledge sources.

Combination of knowledge sources. The rules we will use, will typically be a combination of knowledge from different sources. sources. The previously described example rule is a combination of photometric knowledge (difference in pixel value with its surroundings), traffic knowledge (cars are driving on a lane) and again traffic knowledge (roadmarks usually show up in regular patterns). The four following classes of knowledge sources are not meant to be exhaustive but have to be seen as an indication as to which kind of knowledge sources will be used in the reasoning process.

Photometric knowledge. This is the main knowledge source. We know that for a part of the image to be interpreted as a car, it is very likely that that part has a different pixel signature than its surroundings. We have also made use of the color dominance for a region. It is e.g. plausible to conclude that a red region on the road is very likely to be a car.

Traffic knowledge. By analyzing traffic, we notice that most of the time, cars drive on the same lane. Distances between cars give us good indications as to the speed of these cars. Roadmarks showing the separation between lanes usually come in regular patterns. etc.

Temporal knowledge. The time at which the images were taken, allows us to know the position of the sun. The absolute orientation of the images together with this position gives a clear indication of the direction and length of shades.

Common sense knowledge. It is impossible for two distinct objects to be in the same space-time location as it is impossible for one object to be in two different locations. This knowledge is used in that part of the system where we find the correspondence between cars. The correspondence module is not only considered as a module that at the end of the interpretation process derives which are the cars to be matched. If not all the cars are detected in a part of one of both images, this module can give useful indication as to the probable existence and location of a not yet detected car in one of the images. In doing so we can use its information during the interpretation process.

2.2.4. Reasoning mechanisms. It is not enough to state that to be able to interpret images, one needs to apply rules. Our ultimate goal is of course to have a completely automated system. The main problem to solve is therefore to analyze the nature of the rules and to understand how they have to be applied and when they have to be invoked. We will now briefly describe the reasoning mechanisms we have implemented in our system.

Forward reasoning. We illustrate the notion of forward reasoning through a planning problem. A planning problem involves finding a path from a begin situation to an end situation. For each situation there are a number of possible actions or steps. The problem is then to find the sequence of steps , that lead from this begin to this end situation. A possible approach is to see which steps are possible from the begin situation and to see where each of them leads us to. If one of the new situations is the end situation we found the path and if not we recursively apply the same procedure. The main idea is that we explore forwards. We could however have started from the end situation and considered which steps have lead to this end situation and work our way backwards i.e. backward reasoning. The image interpretation process can be described as a planning problem where the plan consists of finding which rules to apply in order to get from an uninterpreted set of regions to a complete interpretation of these regions. Since we usually do not know this final interpretation, it must be clear that we will only be able to apply a forward reasoning based mechanism. It is important to recognize this in order to have a clear view on our control structure. Forward reasoning is sometimes also described as data driven reasoning since it involves the derivation of new facts (data), based on already known facts. These new fact can again give rise to other facts etc.

Model based reasoning. Vision interpretation systems need a well defined model in order to obtain an optimal interpretation. [1]. One common way to describe this model is by describing the interpreted image as a semantic tree that has to satisfy some consistency rules. At the leafnodes of this tree we will find our regions. An example of a node in our semantic tree could be a node, labeled a car sequence, which would have as children nodes individual cars. These nodes would have at their turn as children nodes one or more regions that constitute these cars. At each interpretation step we will, possibly instantiate a node, and link it with an existing one. We will now briefly describe how bottom up and top down reasoning are defined in a model.

Bottom up reasoning. If we have evidence that a region is a car in our context, then we can instantiate a car node and link it to our region leaf node. In this case we build our semantic tree from the bottom to the top. This reasoning mechanism is therefore called **bottom up**.

Top down reasoning. Sometimes we will already have instantiated a semantic node without knowing to which leaf node it has to be linked. A typical example is the interpretation of a roadmark in one image without it being detected in the other image. Since we know that it cannot disappear in three seconds, we know that it exists in the other image. The roadmark node existing we can now look down in our tree and see which or which combination of leaf nodes best fits our roadmark. Working this way is of course a **top down** approach. (It needs of course some verification that the roadmark is not covered by a car that happens to be on top of the roadmark in a specific image.).

Asserting that there is photometric evidence in the actual context which allows to derive the fact that a car is represented by a certain region, means that one has applied a rule such as described earlier. Such a rule can therefore be applied in as well a bottom up as a top down reasoning scheme.

Testing of hypotheses. A very important and powerful reasoning mechanism that people constantly apply, involves the fact of adding hypotheses to a certain context and to try to verify these hypotheses. An illustrative example involves the hypothesis of the existence of a car in a certain part of the image. Accepting the hypothesis can lead to very unlikely situations in which case we will prefer to say that there is no car in that part of the image (e.g. in order to verify the hypothesis we would need to accept the fact that an other car accelerated so fast that it got out of the image range). We therefore need a system that allows us to make hypotheses, to reason on the basis of them and to allow to come back to our steps when we know that accepting the hypothesis would lead to an inconsistent context.

2.2.5. Control structure. Although it is clear that one needs to apply knowledge based rules, it is not clear when which rule needs to be invoked and which reasoning mechanism needs to be adopted. Expert systems on the market seem to suggest that this is a more or less trivial task. The reality is more complicated. Only very few expert system shells allow for a user determined control structure. A combination of forward reasoning and hypotheses is rarely possible. As an example we cite OPS5 that allows for a forward reasoning scheme but where one has non or little control of the order of invocation of the rules. There is certainly no elegant way to allow for possibly wrong hypotheses. KEE, which is an object oriented expert system shell does allow for forward reasoning and provides some form of hypotheses using the concept of *multiple worlds* . We found that it was difficult to express the control of the invocation of the rules and the reasoning mechanism in KEE. The core control system has therefore been implemented in Prolog.

2.2.6. Representation of context. A question that remains to be answered is how context has been represented in our system. Since we deal with a lot of symbolic information, it seemed obvious that we would keep a symbolic description of what had been interpreted so far. This is partially done that way. The main representation structure for the context however, is an iconic structure. This iconic representation can be seen as a symbolic image. At each image-pixel, symbolic values can be stored. This means that symbolic relations among objects don't need to be stored explicitly but their relation is implicitly defined by the iconic structure. If we want e.g. to check if or which car is driving at a certain location, we don't need to check the complete symbolic list of car descriptions but we can consult our iconic representation to get an immediate answer. This turns out to give a significant computational benefit.

2.3. The implementation.

The segmentation part has been implemented on a VAX-750 running VMS. The Ohlander region finding algorithm has been written in Pascal. The output of this phase is a list of regions with their corresponding features. Other segmentation procedures will be applied later on the image but this will be done in a guided way. The creation of the regions has therefore been implemented as a complete separate step. The regions are then read into KEE as objects. These objects are only used in a very static way. The reason for representing these objects in KEE is the fact that a complete vision interpretation supporting environment has been worked out in KEE on an Explorer II at our lab. The visualization of images and the updating and consultation of the iconic structure has been implemented in this environment. The core control structure has been implemented in Prolog as already stated. It is during the execution of this Prolog program that will be decide which rule to apply in which reasoning mechanism. When data is needed from the regions or from the iconic structure, an interface to KEE via Lisp is used.

3. The results and further work.

In its actual form, the system disposes of all its key features. The final version of the system is expected by mid 89. Some features still need to be implemented. These involve some form of guided resegmentation and the adding of more rules in order to have a system that is able to interpret a wide range of image types. The correspondence module has almost completely been implemented. The integration with the rest of the system still needs to be done in order to use correspondence knowledge during the interpretation process as previously explained.

The actual system allows for approximately 90 % detection rate. But what is even much more important, the false detection rate is virtually 0%. We believe that these results are in themself already encouraging. We are confident however that the addition of more rules and the integration of the above described process will get us close to 100 % and keeping our 0% false detection rate.

4. Conclusion.

The relevance of the automation of the interpretation process can easily be understood by noting that an attempt has been undertaken to do this manually. Because actual results in computer vision clearly show that we are still quite far from human performance in perception, it therefore seemed obvious that it would be better to do the interpretation process manually. It turned out that we, as human interpreters, are indeed better in perceiving cars on these images, but that we are unable to perform as good on the whole image because of lack of concentration. If we then need to focus our attention on two large images, interpret them and make the correspondence between them, it turns out to be virtually impossible to do this task manually. As stated in the result we are aiming at improving our 90 % interpretation results. It is worth noting however that the notion of 100 % complete interpretation is a purely theoretic notion. We had some of the images interpreted by people of our group and it turned out there was no 100% agreement over the result of interpretation. It is also interesting to note that the interpretation process in vision has some very clear correspondence with an other perception problem, namely speech understanding. We believe that it would be possible to apply this general control structure to a broader scope of application domains where there is a need to understand or interpret a signal in some global context.

Acknowledgements

the research is based on images provided by Eurosense-Belfotop.

5. References

[1] Binford, T.O. *Survey of model-based image analysis systems* Int. J. of Robotics Research, 1(1), 1982 pp 18-64.

[2] Fischler, M Firschein, O. *The eye, the brain and the computer* Addison Wesley 1987

[3] McKeown, D. Jr. *Knowledge based aerial photo interpretation* Photogrammetria, 39(1984) pp 91-123.

[4] Ohlander,, R., K.Price, and D.R.Reddy. *Picture segmentation using a recursive region splitting method* CGIP 8,3,December 1979.

Target Cluster Detection in Cluttered Synthetic Aperture Radar Imagery

James G. Landowski and Richard S. Loe
Lockheed Research and Development Laboratories
Department 96-30, Building 251
3251 Hanover St., Palo Alto, California 94304
e-mail: {sun!sunncal,leadsv}!dipl!lando

Abstract

A technique for detecting clusters of objects in noisy, cluttered, moderate resolution imagery is discussed. The algorithm is demonstrated on synthetic aperture radar (SAR) data. The approach is based on the use of a nonlinear spatial highpass or 'antimedian' filter, the complement of the median filter. The filter is coarsely tuned to produce maximum response for structures the size of or smaller than the expected object size. The filter is followed by histogram thresholding and connected region processing. Knowledge about the object's shape and the cluster deployment patterns is then used to eliminate false detections. This detection technique is suitable for any imagery where the objects of interest produce sensor responses that form contiguous regions. False clusters due to edge leakage are discussed an a solution formulated.

1. Introduction

A serious problem in detecting mobile targets is that large areas have to be searched. This imposes stringent requirements on the sensors and the target detection processors. The sensor must have high spatial resolution (small instantaneous field-of-view) in order to accurately detect and classify targets as well as a large field-of-view (FOV) to cover all potential target areas. The processor must examine this high volume of data in near real-time. The classic target detection, feature extraction and classification approach is shown in Figure 1. Target detection is performed after preprocessing the sensor data. The usual approach to detection is to select a distinctive but simple target signature (such as warm targets in FLIR) that can be quickly extracted from the data. Loose constraints are applied to the detection feature to guard against missed detections. The result is that some false alarms are accepted in order to avoid missed detections. In the classical approach the feature extraction and classification stages examine additional (more complicated) target features in order to reject the detection false alarms. Knowledge based approaches have been proposed [1] but these require greater computing resources than the statistical techniques and still require reliable features. Reliable feature extraction requires two to four times higher spatial resolution than detection. Table 1 gives the resolution required by photointerpreters to perform detection, recognition and precise identification for a variety of targets.[2]

One possible means of lessening the strict sensor requirements is to note that some important stationary and mobile targets occur in clusters. Some examples are Tank farms, Surface-to-Air Missile (SAM) batteries and Strategic Relocatable Target (SRT) Battalions as shown in Figure 2. Such clusters are not precisely defined in composition or geometry. However, a SAM battery must contain launchers, radar and a command center as a minimum. Additional vehicles may or may not be present. The battery deployment will depend on local geography and the mission. There are some geometric constraints since the launchers must be connected by power and command cables. Thus the SAM battery will be limited to a maximum size deployment area and the cluster target density is known within certain bounds. Similar constraints apply to the other target types mentioned.

This work was directed towards determining if the classical feature extraction and classification stages could be replaced by a cluster recognition algorithm which takes the target detection output and looks for clusters of targets. (Figure 1 also shows how cluster recognition fits in with the classical target recognition process.) The sensor requirements would then be dictated by the target detection requirements rather than the feature extraction requirements. Previous work in this general area [3] modeled the problem as points in a random field of points and at-

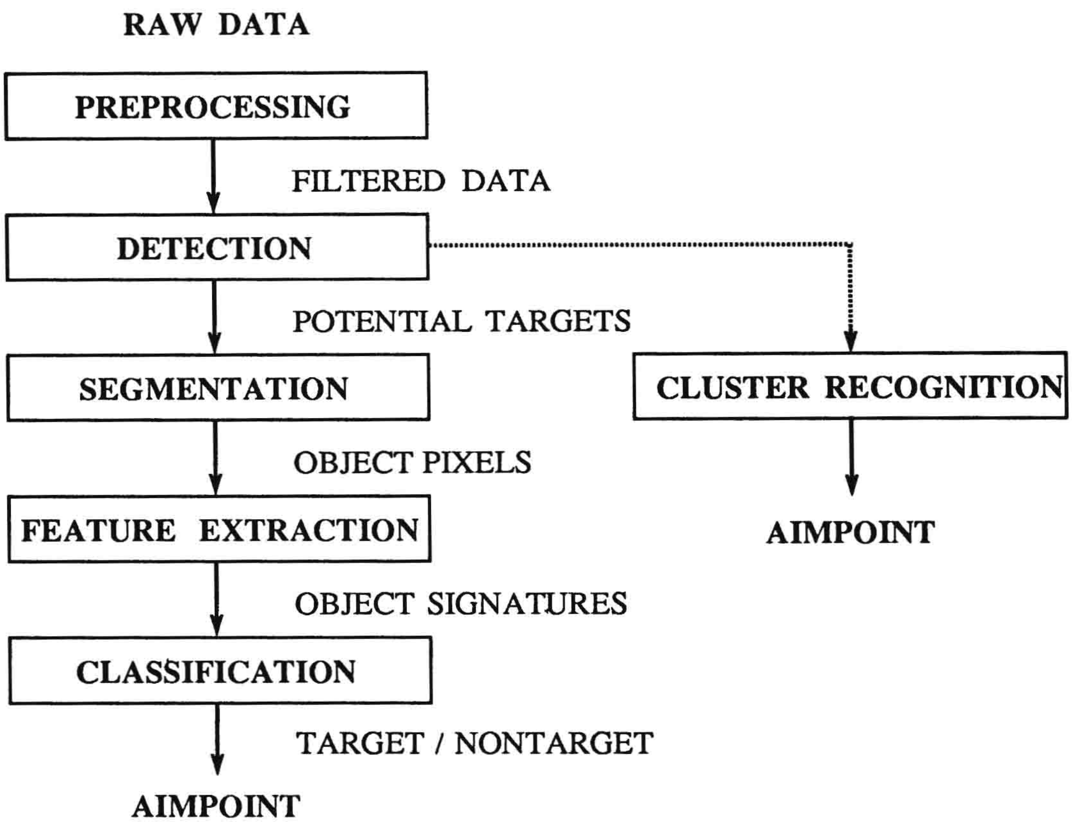

Figure 1. Classic Steps in Target Recognition Compared with Cluster Recognition

tempted to find structures matching a geometric point model of the object of interest. A difficulty with this approach is that the geometric model is too strict a requirement since deployment patterns in many cases are bounded in extent but arbitrary in geometry.

2. Approach

The detection process is outlined in Figure 3. It is based on the observation that SAR produces samples of known size on the ground. Since the target size is known, one can apply a size filter to eliminate all objects of incorrect size. The size filter is based on an antimedian filter, the complement of the median filter[4]. A median filter examines the pixel values within a window centered on the current pixel being processed, ranks the pixel values by intensity and replaces the current pixel by the pixel in the middle of the ranking (i.e. the median value). This eliminates detail smaller than about half the area of the filter window while retaining the large image detail. The antimedian filter is formed as in equation 1 by subtracting the results of the median filter from the original image. The antimedian filter performs a highpass filtering function that eliminates image detail larger than the filter window size without degrading the shape of the small artifacts. Thus if the target size is known and is small, an antimedian filter can be designed to emphasize target sized objects. Segmenting the targets from the background is performed by computing the histogram of the results of the antimedian filter. For this work, the graylevel intensities occupying the top 10 percent of the histogram are set to 1 (detection) and the rest of the values are set to zero. The histogram cutoff is establish by making an assumption on the expected target density and size in the image. The bina-